



# MANUALE WEBAPI

**20/07/2022 v1.6**

## Sommario

<b>1. Introduzione</b>	<b>2</b>
1.1. Cosa è	2
1.2. Glossario	2
1.3. Come si usa	2
<b>2. Formato della http request</b>	<b>3</b>
2.1. Installazioni locali	3
2.2. Installazioni LIVE	5
<b>3. Comandi e scambio dei dati</b>	<b>6</b>
3.1. Tipo di comandi e spazio dei nomi	6
3.2. Metodo GET	7
3.3. Metodo POST	8
3.4. Metodo PUT	9
3.5. Metodo DELETE	9
3.6. Entità ricerca	9
3.7. Entità allegati	10
3.8. Entità MyDB	11
3.9. Entità liste-prelievo	12
3.10. Entità ubicazioni	15
3.11. Trasformazione documenti	16
3.12. Webapi per la richiesta di servizi	18
3.13. Riassunto dei comandi RESTful	20
<b>4. Accesso ai dati del gestionale</b>	<b>21</b>
4.1. Esempio gestione ordini clienti	21
<b>5. CHANGELOG</b>	<b>24</b>
V1.6: dalla versione gestionale 2022G – v824	24
v1.5.2: dalla versione gestionale 2022D – v821	24
v1.5.1: dalla versione gestionale 2022C2 – v820b	25
V1.5: dalla versione gestionale 2022C1 – v820a	26
V1.4: dalla versione gestionale 2022B4 – v819d	26
V1.3: dalla versione gestionale 2022B – v819	26
V1.2: dalla versione gestionale 2022A – v818	27
V1.1 : dalla versione gestionale 2021I – v816	27
<b>6. Allegati</b>	<b>29</b>
Allegato A	29

# 1. INTRODUZIONE

## 1.1. COSA È

WebApi è, come suggerisce il nome, una collezione di API Passepartout basate su tecnologia Web tramite protocollo https. Il servizio è fruibile utilizzando richieste https secondo lo stile architetturale RESTful: **R**epresentational **S**tate **T**ransfer. Sono implementati tutti i metodi CRUD (**C**reate, **R**ead, **U**ppdate, **D**eleTe) per mezzo dei metodi http **POST**, **GET**, **PUT**, **DELETE**.

Secondo il paradigma REST il servizio è fortemente orientato alle entità (gli archivi del gestionale) le quali saranno raggiungibili mediante path specifici.

## 1.2. GLOSSARIO

Di seguito la terminologia ed il significato dei termini utilizzati in questo manuale.

- *risorsa*: oggetto presente sul gestionale Mexal/Passcom su cui possono essere effettuate operazioni
- *entità*: risorsa. Entità e risorsa verranno trattati come sinonimi all'interno di questo manuale
- *http*: **H**ypertext **T**ransfer **P**rotocol
- *https*: **H**ypertext **T**ransfer **P**rotocol over **S**ecure **S**ocket **L**ayer. Nel corso di questo manuale potrà capitare di riferirsi indistintamente ad http o https per definire alcuni concetti. WebAPI gestisce solo richieste tramite protocollo sicuro https.
- *http request*: azione (POST, GET, PUT, DELETE) fatta su una risorsa identificata da un dato url
- *header*: parte della *http request* in cui vengono impostati metadati necessari al completamento della richiesta stessa
- *body*: corpo di una *http request* in cui sarà presente il dato da scambiare
- *query string*: è la parte di un URL che contiene dei dati da passare in input ad un servizio
- *JSON*: **J**ava**S**cript **O**bject **N**otation, è un formato standard adatto all'interscambio di dati tra applicazioni client/server
- *installazione*: ci si riferisce al programma gestionale Mexal/Passcom
- *ambiente live*: installazione presente in server farm Passepartout (cloud)
- *ambiente locale*: installazione presente su un server locale alla rete del cliente (hausing)
- *front-end*: parte di un programma responsabile dell'acquisizione dei dati di ingresso
- *back-end*: parte di un programma che elabora i dati acquisiti
- *cookie*: è una stringa di testo di piccole dimensioni inviata da un web server ad un web client e poi rimandata indietro dal client al server; tipicamente è utilizzata per implementare meccanismi di identificazione di un client presso un server

## 1.3. COME SI USA

WebApi è disponibile sia in ambiente locale che live dopo aver attivato la suite MDS tramite YouPass. L'attivazione del servizio abiliterà in Mexal/Passcom un gruppo particolare di utenti "Servizi WebApi". Solo gli utenti appartenenti a gruppi di questo tipo potranno utilizzare solo ed esclusivamente i servizi WebApi.



WebApi locale: è possibile accedere su installazione locale specificando l'indirizzo del server più la porta su cui è in ascolto il servizio. Il servizio deve essere abilitato lato server Mexal/Passcom tramite il menu *Servizi->Configurazioni->Configurazione moduli->WebApi*. In questa videata sarà possibile verificare la corretta abilitazione del modulo e la porta assegnata al servizio. L'accesso sarà possibile specificando nell'header della richiesta https le credenziali di accesso più altri parametri per identificare l'azienda e la data di esercizio.

WebApi Live: in ambiente live, se attivato da codice contratto, il servizio è già disponibile e raggiungibile tramite l'indirizzo <https://services.passepartout.cloud/>. In questo caso l'autenticazione avviene sempre tramite credenziali di accesso nell'header della richiesta https (come per le installazioni locali) ma usando come *base address* <https://services.passepartout.cloud/webapi/risorse> ([vedi sezione dedicata](#))

Il gruppo Servizi WebAPI gestisce un numero massimo di 3 utenti: ogni utente del gruppo Servizi WebApi avrà a disposizione un pool di massimo 5 servizi contemporanei con un time-out di 5 minuti per ogni servizio. Questo vuol dire che lo stesso utente WebAPI potrà accedere contemporaneamente solo da 5 client differenti (o 5 richieste concorrenti dallo stesso client), le successive richieste verranno rifiutate fino a che uno dei servizi del pool non si libera.

## 2. FORMATO DELLA HTTP REQUEST

### 2.1. INSTALLAZIONI LOCALI

Per utilizzare WebApi in installazioni locali è necessario attivare il modulo dal menu Servizi - Configurazioni - Configurazione moduli – WebApi.



Le informazioni necessarie per l'accesso sono:

URL = <indirizzo\_server>  
 PORTA = <porta\_server\_https>  
 USER = <utente\_di\_mexal\_passcom>  
 PWD = <password\_utente\_mexal\_passcom>

Se la richiesta prevede l'utilizzo di un comando che coinvolge una determinata azienda, è necessario fornire anche le seguenti informazioni:

AZIE = <azienda>  
 SOTTOAZ = <sottoazienda> (opzionale)  
 ANNO\_AZ = <anno\_di\_apertura>  
 MAGAZZINO=<numero\_magazzino\_default> (opzionale)

**NOTA: se non viene utilizzato il numero di magazzino, tutte le operazioni che coinvolgono il magazzino utilizzeranno come magazzino di default il numero 1.**

Se sull'installazione locale è attivo anche il controllo delle credenziali di sistema operativo è necessario utilizzare anche queste credenziali

USER\_SYS = <utente\_di\_sistema>  
 PWD\_SYS = <password\_utente\_di\_sistema>

L'indirizzo per raggiungere WebApi in locale dovrà essere formato in questo modo:

**https://URL:PORTA/webapi/risorse/**

Esempi:

**PUT** https://miodominio:9004/webapi/risorse/clienti/codice\_cliente (aggiorna il cliente: dati nel corpo della richiesta)

**POST** https://miodominio:9004/webapi/risorse/articoli (crea un nuovo articolo: dati nel corpo della richiesta)

**GET** https://miodominio:9004/webapi/risorse/articoli (legge tutti gli articoli)

**DELETE** https://miodominio:9004/webapi/risorse/articoli/codice\_articolo (cancella un articolo)

Le credenziali di accesso USER e PWD devono essere formattate in questo modo:

- concatenare USER e PWD formando la seguente stringa: "USER:PWD" es. "ADMIN:MEXAL"
- fare un encoding in base64 della stringa ottenuta al passo precedente

Nell'header della richiesta https dovranno poi essere inserite le informazioni ottenute dai passi precedenti in questo modo:

```
'Authorization': 'Passepartout <credenziali_b64>'
'Content-type': 'application/json'
'Coordinate-Gestionale': 'Azienda=<AZIE> SottoAzienda=<SOTTOAZ> Anno=<ANNO_AZ> Magazzino=<MAGAZZINO>'
```

Nel caso l'installazione gestisca anche l'ulteriore autenticazione tramite credenziali di sistema operativo (login=1) devono essere aggiunte nella chiave 'Authorization' codificate in base64 analogamente a quanto fatto per le credenziali dell'utente Mexal/Passcom.

L'header in questo caso avrà una struttura di questo tipo:

```
'Authorization': 'Passepartout <credenziali_b64> <credenziali_sistema_b64>'
'Content-type': 'application/json'
'Coordinate-Gestionale': 'Azienda=<AZIE> SottoAzienda=<SOTTOAZ> Anno=<ANNO_AZ> Magazzino=<MAGAZZINO>'
```

**Il Content-type (opzionale solo per comandi GET e DELETE) prevede come charset di default utf-8 e non è necessario specificarlo. Se però si vuole specificare anche il charset è necessario che sia utf-8: 'Content-type': 'application/json;charset=utf-8'. Qualsiasi altro tipo di charset verrà rifiutato con un errore.**

Di seguito due esempi di come deve essere formattato un header di una richiesta https su installazioni locali.

Caso di autenticazione con solo le credenziali dell'utente dell'installazione:

```
{
  'Authorization': 'Passepartout QURNSU46TUVYQUw=',
  'Content-type': 'application/json',
  'Coordinate-Gestionale': 'Azienda=DEM Anno=2021 Magazzino=2'
}
```

Caso di autenticazione in cui sono necessarie anche le credenziali di sistema operativo (login=1):

```
{
  'Authorization': 'Passepartout QURNSU46TUVYQUw= QURNSU46QURNSU4=',
  'Content-type': 'application/json',
  'Coordinate-Gestionale': 'Azienda=DEM Anno=2021'
}
```

**Nel caso di installazioni locali, l'accesso alle risorse tramite protocollo sicuro https avviene utilizzando un certificato autoprodotta (self-signed). Le librerie utilizzate dall'applicazione client dovranno quindi poter gestire questo caso.**

**In alternativa, se non fosse possibile utilizzare i certificati self-signed, è necessario dotare l'installazione di certificati validi (non self signed). Questo è possibile farlo inserendo i certificati tramite il menu Servizi - Configurazioni - Configurazione moduli - WebApi nella sezione Configurazione Server (in modo del tutto analogo a quanto avviene per WebDesk).**

## 2.2. INSTALLAZIONI LIVE

Per poter accedere in ambiente Live, i parametri necessari per l'accesso sono i seguenti:

<https://services.paspartout.cloud/webapi/risorse>

```
DOMINIO = <dominio_della_installazione_live>
USER = <utente_di_mexal_passcom>
PWD=<password_utente_mexal_passcom>
```

Se la richiesta prevede l'utilizzo di un comando che coinvolge una determinata azienda, è necessario fornire anche le seguenti informazioni:

```
AZIE = <azienda>
SOTTOAZ = <sottoazienda> (opzionale)
ANNO_AZ = <anno_di_apertura>
MAGAZZINO=<numero_magazzino_default> (opzionale)
```

Come per le installazioni locali, le credenziali di accesso USER e PWD devono essere formattate in questo modo:

- concatenare USER e PWD formando la seguente stringa: "USER:PWD" (es. "ADMIN:ADMIN")
- fare un encoding in base64 delle credenziali

Nell'header della richiesta https dovranno essere inserite le informazioni ottenute dai passi precedenti, dove rispetto alle installazioni locali nell'elemento Authorization va specificato anche il dominio nella forma *Dominio*=<MIO\_DOMINIO>

```
'Authorization': 'Paspartout <credenziali_b64> Dominio=<DOMINIO>'
'Content-type': 'application/json'
'Coordinate-Gestionale': 'Azienda=<AZIE> SottoAzienda=<SOTTOAZ> Anno=<ANNO_AZ>'
```

Di seguito un esempio di come deve essere formattato un header di una richiesta https in ambiente live.

```
{
  'Authorization': 'Paspartout QURNSA46QQ== Dominio=MIODOMINIO',
  'Content-type': 'application/json',
  'Coordinate-Gestionale': 'Azienda=DEM Anno=2021'
}
```

## 3. COMANDI E SCAMBIO DEI DATI

Tutti gli esempi e le descrizioni dei comandi da questo punto in poi avranno lo stesso formato, sia per installazioni locali che live, lasciando sottointeso il *base address*. Per semplicità quindi tratteremo il path in questo modo:

```
https://base_address/webapi/risorse/<nome_risorsa>
```

dove *base\_address* potrà essere:

- <indirizzo\_server>:<porta\_https> nel caso di installazioni locali
- services.passepartout.cloud nel caso di installazioni live

Per tutti i casi di autenticazione citati, l'interazione con Mexal/Passcom avviene mediante i seguenti metodi http:

**POST:** utilizzato esclusivamente per creare una nuova entità (es. creazione di un nuovo cliente)

**GET:** utilizzato per leggere una entità o una collezione di entità (es. la lista di tutti i clienti)

**PUT:** utilizzato esclusivamente per aggiornare una entità (es. modifica di una descrizione di un articolo)

**DELETE:** utilizzato per cancellare una entità (es. eliminazione di un cliente)

I metodi POST e PUT prevedono anche un *payload* contenuto nel *body* della richiesta https in formato JSON in cui saranno inseriti i dati da inserire o aggiornare, mentre per i comandi di tipo GET e DELETE un eventuale contenuto viene ignorato.

### 3.1. TIPO DI COMANDI E SPAZIO DEI NOMI

Mexal/Passcom prevede due tipi di comandi:

- Comandi che richiedono necessariamente l'apertura di una azienda in un dato anno
- Comandi sovra-aziendali che possono essere eseguiti anche senza l'accesso ad una particolare azienda

Nel primo caso, nell'header di tutte le richieste https devono essere specificati anche i parametri di accesso alla particolare azienda (o sottoazienda), come indicato nel [capitolo 2](#).

Nel secondo caso tali parametri possono essere omessi in quanto non verranno controllati dal servizio WebAPI.

Tutti i nomi sono in alfabeto minuscolo senza indicazione sul formato del dato ed eventuali nomi composti saranno separati dal carattere *underscore* (\_).

I campi di tipo *array* possono arrivare ad un massimo di 3 livelli. Ogni elemento di un campo di tipo array è identificato a sua volta da un elemento di tipo array in cui sono specificati gli indici dei vari livelli ed il valore in ultima posizione.

**Si tratta quindi di array misti in cui i primi valori sono sempre interi (indici), l'ultimo valore può essere intero o stringa a seconda del tipo di dato.**

Ad esempio per un campo array di 3 livelli, per identificare un elemento, la struttura sarà di questo tipo:

```
"campo_array": [
  [1,2,1,"a"], // valore "a" in posizione 1 nel primo livello, 2 nel secondo e 1 nel terzo
  [2,1,4,"b"] // valore "b" in posizione 2 nel primo livello, 1 nel secondo e 4 nel terzo
]
```

Per i nomi delle variabili, il tipo di dato ed il significato usare il [metodo GET](#) con il parametro *info=true* nella query string.

## 3.2. METODO GET

Secondo l'architettura REST i comandi di tipo GET non devono specificare alcun contenuto nel corpo della richiesta http. In ogni caso anche se presente deve essere ignorato.

Per capire quali sono le entità accessibili è disponibile una entità *HELP* utilizzabile con metodo *GET* che restituisce un JSON con tutte le entità ed i metodi disponibili.

**GET https://base\_address/webapi/risorse/help**

Verrà tornato un JSON con l'elenco di tutti gli end-point gestiti. Di default verranno restituiti il path, la descrizione ed il metodo disponibile. Se si vogliono ulteriori informazioni è possibile usare il parametro *extended* in query string

**GET https://base\_address/webapi/risorse/help?extended=true**

```
{
  "next": true/false,           // specifica se la risorsa gestisce il parametro next
  "filter": true/false,        // specifica se la risorsa gestisce i filtri
  "regexp": "<path_risorsa>",   // path e la regular expression associata alla risorsa
  "descrizione": "<descrizione_risorsa>", // descrizione della risorsa
  "coordinate": true/false,    // specifica se per l'accesso alla risorsa servono le coordinate aziendali
  "versione": "<versione_gestionale>", // specifica da che versione contenitore (Mexal/Passcom) è disponibile la risorsa
  "method": "<metodo_http>",    // specifica il metodo http disponibile per la
  "max": true/false,           // specifica se la risorsa gestisce il parametro max
  "chiavi": [ <elenco_chiavi> ], // specifica eventuali chiavi per referenziare gli elementi di una risorsa
  "location": true/false,     // specifica se la risorsa ritorna un tag location nell'header
  "fields": true/false        // specifica se la risorsa gestisce il parametro fields
}
```

Facendo GET su un'entità senza specificare altri parametri, il JSON di risposta conterrà tutti i record con tutti i campi dell'archivio.

**GET https://base\_address/webapi/risorse/articoli**

Se invece si vuole recuperare il singolo record bisogna specificare l'ID/Codice della particolare entità

**GET https://base\_address/webapi/risorse/articoli/11AA55**

**Nota: la differenza sostanziale tra i comandi GET che tornano una collezione di elementi rispetto a quelli che tornano un singolo elemento sta nel fatto che nel primo caso il risultato potrebbe avere alcuni campi in meno. Inoltre gli array sono tutti al massimo di 2 dimensioni, mentre nel secondo caso gli array possono avere fino a un massimo di 3 dimensioni. Per i dettagli sui campi disponibili usare *info=true* in query string.**

È possibile operare dei filtri usando le *query string*.

Per specificare i soli campi che si vuole vengano restituiti bisogna utilizzare l'attributo *fields*

**GET https://base\_address/webapi/risorse/articoli/11AA55?fields=codice,descrizione**

**Nota: è sempre consigliato estrarre i soli campi necessari in modo da velocizzare l'elaborazione lato server, soprattutto nel caso che la risorsa da gestire sia molto corposa.**

È possibile specificare il numero massimo di record restituiti nel caso si richieda una collezione di risorse. In *query string* sarà necessario utilizzare l'attributo *max* (massimo 100)

**GET https://base\_address/webapi/risorse/articoli?fields=codice,descrizione&max=100**

La risposta, oltre ai record letti, in questo caso contiene anche un elemento che identifica quale sarà il successivo record da cui partire per eventuali successive letture.

```
{
  [
    {
      "codice": "123ABC",
      "descrizione": "Pacco pignoni"
    },
    {
      "codice": "123456",
      "descrizione": "Cambio SRAM"
    },
    {
      "codice": "ABCDEF",
      "descrizione": "Gomma MAXISS ARDENT"
    }
  ],
  "next": "0F45AD69"
}
```

Di default il metodo GET legge al massimo 100 record (come se fosse impostato `max=100`). Per archivi più lunghi di 100 record verrà quindi sempre restituito un elemento `next` indicante il prossimo record da cui partire per completare la lettura.

È possibile specificare il numero di record da cui partire se è stato restituito un `next` in un precedente metodo GET

**GET** `https://base_address/webapi/risorse/articoli?fields=codice,descrizione&next=0F45AD69`

Per ogni risorsa è possibile ottenere un elenco di tutti i campi disponibili ed il loro significato. È sufficiente specificare nel path della query string l'attributo `info=true`, in questo caso tutte le altre query string (se presenti) verranno ignorate.

**GET** `https://base_address/webapi/risorse/articoli?info=true`

Nel caso di GET su una collezione di entità è possibile impostare anche filtri più evoluti oltre all'utilizzo di quelli specificati in *query string* (`fields`, `max` e `next`).

I filtri dovranno essere impostati nel corpo della richiesta in formato JSON. Dato però che, come precedentemente indicato, il metodo GET non prevede l'utilizzo di alcun contenuto nel body della http request, i filtri saranno utilizzati sfruttando il metodo POST.

Ogni entità ha a disposizione una risorsa denominata *ricerca* su cui sarà possibile specificare opportuni filtri nel body della request.

**POST** `https://base_address/webapi/risorse/articoli/ricerca` ([vedi entità RICERCA](#))

### 3.3. METODO POST

Il metodo POST deve essere utilizzato esclusivamente per la creazione di nuove entità. Nel path non deve essere mai specificato il codice della risorsa da creare. Nel body della richiesta https deve essere presente un JSON con i dati da inserire. Nel caso debba essere specificato un codice per la creazione della risorsa (vedi anagrafica articoli) bisogna aggiungerlo nel body della richiesta https.

Se la creazione avviene con successo verrà tornato un *http status code 201 (Created)*.

**POST** `https://base_address/webapi/risorse/clienti`

```
{
  "codice": "501.AUTO", //codice creato in automatico sul mastro indicato
  "ragione_sociale": "Cliente di prova",
  "tp_nazionalita": "I",
  // altri campi obbligatori
}
```

Creazione con codice specificato dall'utente

**POST** `https://base_address/webapi/risorse/articoli`

```
{
```

```

"codice": "ARTICOLO1",
"descrizione": "Articolo di prova"
//altri campi obbligatori
}
    
```

Il response di una *http request* conterrà nell'header della risposta un tag *location* comprendente il path della nuova entità appena creata.

### Creazione di un articolo

#### POST [https://base\\_address/webapi/risorse/articoli](https://base_address/webapi/risorse/articoli)

Body della richiesta https:

```

{
"codice": "ARTICOLO1",
"descrizione": "Articolo di prova"
//altri campi obbligatori
}
    
```

Header della risposta https:

```

{
// altri tag
'Content-type': 'application/json; charset=utf-8',
'Location': 'https://base_address/webapi/risorse/articoli/ARTICOLO1',
}
    
```

## 3.4. METODO PUT

Il metodo PUT deve essere utilizzato per modificare/aggiornare delle risorse già esistenti. In questo caso è obbligatorio specificare sempre il codice della risorsa nel PATH, mentre i dati devono essere specificati nel body della *http request*. Non è previsto alcun contenuto nel *body* della risposta https. In caso di successo verrà solamente restituito un *http status code 204 (No Content)*.

#### PUT [https://base\\_address/webapi/risorse/articoli/11AA55](https://base_address/webapi/risorse/articoli/11AA55)

Body della richiesta https:

```

{
"descrizione": "Descrizione modificata"
//altri campi
}
    
```

## 3.5. METODO DELETE

Il metodo DELETE deve essere utilizzato per eliminare una risorsa esistente. In questo caso è obbligatorio specificare sempre il codice della risorsa nel PATH. Non è previsto alcun contenuto nel *body* della risposta https. In caso di successo verrà solamente restituito un *http status code 204 (No Content)*.

#### DELETE [https://base\\_address/webapi/risorse/articoli/11AA55](https://base_address/webapi/risorse/articoli/11AA55)

## 3.6. ENTITÀ RICERCA

La risorsa *ricerca* è una particolare risorsa comune a tutti i gruppi di risorse (es. clienti, articoli, documenti, etc.).

Questa particolare risorsa può essere utilizzata solo mediante metodo POST. Il significato è: "crea una ricerca su questa collezione di risorse". In questo modo è possibile ottenere un risultato come se fosse una GET in cui sono specificati filtri particolari nel *body* della *request* oltre quelli specificati in query string. Come per il comando GET è possibile utilizzare i filtri in query string (*fields, max e next*).

I filtri devono essere specificati secondo questo pattern

```

{
    
```

```

"filtri":[
  {
    "campo": "<nome_campo>",
    "indice1": "<indice1_campo_array>", // facoltativo: serve nel caso il campo sia un array. Nel caso di liste, gli array sono di
                                         massimo 2 livelli quindi possono essere specificati solo 2 indici
    "indice2": "<indice2_campo_array>",
    "condizione": "=", "<=>", ">=", "<>", "contiene",
    "case_insensitive": true/false // default: false
    "valore": "<valore da testare>"
  },
  // altri elementi in AND
]
}

```

Nel caso la condizione sia "contiene", può essere indicato anche se la ricerca deve essere case sensitive o meno specificando il parametro "case\_insensitive" (default: false)

**POST [https://base\\_address/webapi/risorse/articoli/ricerca?fields=codice,descrizione](https://base_address/webapi/risorse/articoli/ricerca?fields=codice,descrizione)**

Esempio body della richiesta https

```

{
  "filtri":[
    {
      "campo": "tipologia",
      "condizione": "=",
      "valore": "A"
    },
    {
      "campo": "cos_ult",
      "condizione": ">",
      "valore": 100
    }
  ]
}

```

I filtri sono tutti in logica AND tra loro. Nell'esempio verranno restituiti solo gli articoli di tipo merce e costo ultimo anno corrente maggiore di 100 e solo i campi codice e descrizione.

## 3.7. ENTITÀ ALLEGATI

Solo per gli archivi che hanno allegati (es. immagini articoli) è disponibile una entità dedicata per l'accesso ai documenti.

**GET [https://base\\_address/webapi/risorse/<categoria\\_risorsa>/<codice\\_risorsa>/allegati/<tipo\\_allegato>](https://base_address/webapi/risorse/<categoria_risorsa>/<codice_risorsa>/allegati/<tipo_allegato>)**

Dove tipo allegato è il tipo di file che si vuole referenziare. Nell'header della risposta sarà specificato il content-type corretto per il file restituito (ad esempio image/jpeg per le immagini).

Esempio di lettura delle tre immagini articolo (icona, catalogo, immagine):

Letture dell'icona di un articolo

**GET [https://base\\_address/webapi/risorse/articoli/FUELEX97/allegati/icona](https://base_address/webapi/risorse/articoli/FUELEX97/allegati/icona)**

Letture dell'immagine di catalogo

**GET [https://base\\_address/webapi/risorse/articoli/FUELEX97/allegati/immagine-catalogo](https://base_address/webapi/risorse/articoli/FUELEX97/allegati/immagine-catalogo)**

Letture dell'immagine articolo

**GET [https://base\\_address/webapi/risorse/articoli/FUELEX97/allegati/immagine](https://base_address/webapi/risorse/articoli/FUELEX97/allegati/immagine)**

### 3.8. ENTITÀ MYDB

Per quanto riguarda l'accesso agli archivi MyDB sono disponibili tutti i metodi REST con la stessa logica delle altre entità, ma vale la pena entrare più nel dettaglio in quando si tratta di una risorsa particolare.

La risorsa mydb non è altro che un contenitore (un po' come succede con la risorsa documenti) e quindi non è direttamente accessibile: è necessario specificare sempre nel path anche l'archivio mydb sul quale si intende operare.

Il path sarà quindi di questo tipo

**https://base\_address/webapi/risorse/mydb/MIA\_APP@mydb**

dove MIA\_APP è l'app Passbuilder in cui è presente l'archivio MyDB su cui voglio operare, e mydb il nome dell'archivio MyDB

Facendo una GET verrà restituita una lista di tutti i record dell'archivio MyDB indicato, mentre specificando l'id verrà restituito il singolo record

Lista record

**GET https://base\_address/webapi/risorse/mydb/MIA\_APP@mydb**

Letture del record 42

**GET https://base\_address/webapi/risorse/mydb/MIA\_APP@mydb/42**

In entrambi i casi i dati restituiti avranno un formato di questo tipo (per le liste ovviamente saranno array di oggetti):

```
{
  "nome_archivio": "MIA_APP@mydb",
  "sigla_doc": "",
  "id": 42,
  "numero_campi": 5,
  "dati_campi": [], // array contenente i valori dei campi dell'archivio MyDB
  "tipi_campi": [], // array contenente il tipo dei campi
  "etichette_campi": [], // array contenente le etichette dei campi MyDB
  "riportabil_doc": [], // array contenente i flag sui campi riportabili
  "annullato": "",
  "bloccato": "",
  "utente_ult_mod": "198187508",
  "data_ult_mod": "20211217 113408"
}
```

Se si vuole interrogare un archivio MyDB collegato ad un documento va specificata anche la sigla documento.

Nel caso di richiesta della lista di tutti i record collegati ad un particolare tipo di documenti (es. OC) bisogna specificare la sigla documento come entità

**GET https://base\_address/webapi/risorse/mydb/MIA\_APP@mydb/OC**

Nel caso di singolo record l'id del record va specificato come sotto entità

**GET https://base\_address/webapi/risorse/mydb/MIA\_APP@mydb/OC/42**

I dati restituiti avranno valorizzato anche l'elemento "sigla\_doc"

```
{
  "nome_archivio": "MIA_APP@mydb",
  "sigla_doc": "OC",
  "id": 42,
  "numero_campi": 2,
  "dati_campi": [],
  "tipi_campi": []
}
```

```

"etichette_campi": [],
"riportabil_doc": [],
"annullato": "",
"bloccato": "",
"utente_ult_mod": "626892910",
"data_ult_mod": "20211217 122426"
}

```

I restanti metodi (POST,PUT,DELETE) seguono la solita logica.

Esempio di creazione di un nuovo record su un MYDB di un campo collegato ai documenti OC

**POST [https://base\\_address/webapi/risorse/mydb/MIA\\_APP@mydb/OC](https://base_address/webapi/risorse/mydb/MIA_APP@mydb/OC)**

Body della request:

```

{
  "id": 0,
  "numero_campi": 2,
  "dati_campi": [
    [
      1,
      "C000OC001.000008" // codice del documento (vedi manuale MYDB)
    ],
    [
      2,
      "valore_campo"
    ]
  ]
}

```

**Nota: l'id del record deve essere presente e valorizzato a 0 dentro il body della request**

### 3.9. ENTITÀ LISTE-PRELIEVO

L'entità liste di prelievo è una struttura master-detail, per cui è presente una testata ed n righe ad essa associata (come per i documenti). Sono quindi presenti end-point per accedere alla lista nel suo complesso ed altri per accedere solo alle righe:

- liste

**[https://base\\_address/webapi/risorse/liste-prelievo](https://base_address/webapi/risorse/liste-prelievo)**

- righe

**[https://base\\_address/webapi/risorse/liste-prelievo/righe](https://base_address/webapi/risorse/liste-prelievo/righe)**

Come per i documenti, solo la singola lista può essere modificata (PUT) specificando tutte le righe a cui si vogliono apportare modifiche. Ogni riga d'ordine è identificata dalla chiave composta da sigla, serie, numero e id\_riga.

La peculiarità di questa entità risiede nel fatto che è possibile inserire un altro livello di dettaglio: le righe di picking. Per ogni riga della lista è infatti possibile "creare" ulteriori righe di prelievo (picking) ad essa associata.

Per ogni record (riga) della lista il campo "id\_riga" identifica a quale riga d'ordine è associato. Se la riga è una riga di picking l'elemento "prog\_riga" identifica il progressivo di picking relativo alla riga d'ordine, altrimenti se per la riga non è presente l'elemento "prog\_riga" si intende che si tratta della riga d'ordine.

Nel caso si voglia inserire una nuova riga di picking, è sufficiente fare una PUT referenziando la riga d'ordine su cui si vuole agire ed impostando il progressivo di riga a -1. In questo caso è necessario specificare, oltre ai dati di chiave, solo le quantità.

#### ESEMPI DI REQUEST SU LISTE-PRELIEVO

Letture delle testate di tutte le liste di prelievo

**GET [https://base\\_address/webapi/risorse/liste-prelievo](https://base_address/webapi/risorse/liste-prelievo)**

Letture del dettaglio (testata+righe) di una lista di prelievo

**GET** [https://base\\_address/webapi/risorse/liste-prelievo/5](https://base_address/webapi/risorse/liste-prelievo/5)

Ricerca di una lista di prelievo (solo testate, filtri nel JSON)

**POST** [https://base\\_address/webapi/risorse/liste-prelievo/ricerca](https://base_address/webapi/risorse/liste-prelievo/ricerca)

Modifica di una lista di prelievo (testate + righe)

**PUT** [https://base\\_address/webapi/risorse/liste-prelievo/ricerca](https://base_address/webapi/risorse/liste-prelievo/ricerca)

Con la PUT è possibile modificare anche i soli campi di testata omettendo le righe.

Letture di tutte le righe di tutte le liste di prelievo

**GET** [https://base\\_address/webapi/risorse/liste-prelievo/righe](https://base_address/webapi/risorse/liste-prelievo/righe)

Ricerca di specifiche righe (impostare il filtro nel JSON)

**POST** [https://base\\_address/webapi/risorse/liste-prelievo/righe/ricerca](https://base_address/webapi/risorse/liste-prelievo/righe/ricerca)

Cancellazione di una lista (solo id lista) o singole righe (parametri di filtro in query string)

**DELETE** [https://base\\_address/webapi/risorse/liste-prelievo/5?sigla=OC,serie=1,numero=339,id\\_riga=4,prog\\_riga=2](https://base_address/webapi/risorse/liste-prelievo/5?sigla=OC,serie=1,numero=339,id_riga=4,prog_riga=2)

## ESEMPIO DI INSERIMENTO DI UNA RIGA DI PICKING

Consideriamo la seguente GET sulla lista di prelievo 4 avente una sola riga d'ordine

**GET** [https://base\\_address/webapi/risorse/liste-prelievo/4](https://base_address/webapi/risorse/liste-prelievo/4)

```
{
  "id": 4,
  "data_creazione": "20220610",
  "stato": "C",
  "descrizione": "Lista Prelievo del 10/06/2022",
  "tipo": "V",
  "sigla": [
    [
      1,
      "OC"
    ]
  ],
  "sottoazienda": [],
  "serie": [
    [
      1,
      1
    ]
  ],
  "numero": [
    [
      1,
      330
    ]
  ],
  "id_riga": [
    [

```

```
    1,
    1
  ]
},
"prog_riga": [],
"codice_articolo": [
  [
    1,
    "ATT001"
  ]
],
"data_documento": [
  [
    1,
    "20220302"
  ]
],
"nr_riga_art": [
  [
    1,
    49
  ]
],
"id_magazzino": [
  [
    1,
    1
  ]
],
"id_ubicazione": [],
"dt_sca_doc": [],
"tp_um_articolo": [
  [
    1,
    "1"
  ]
],
"um": [
  [
    1,
    "PZ"
  ]
],
"nr_colli": [],
"quantita": [
  [
    1,
    1.0
  ]
],
"decimali_qta": [
  [
    1,
    2
  ]
],
"coeff_conv": [],
"id_lotto": [],
"cod_taglia": [],
"numero_taglia": [],
"qta_taglia": [],
"id_alias": [],
"cod_conto": [
  [
    1,
    "501.00022"
  ]
],
"cod_agente": [
  [
    1,
    "601.00006"
  ]
]
```

```

    ]
  ],
  "cod_vettore": [],
  "conto_ind_sped": [],
  "ind_spedizione": [],
  "nr_righe_lista": 0
}
    
```

Per aggiungere una riga di picking relativa alla riga d'ordine è necessario fare una PUT indicando, oltre alla chiave per identificare la riga (silga+serie+numero+id\_riga), la quantità di picking (in questo caso "quantita", se l'articolo è gestito a colli o taglie è necessario utilizzare i campi dedicati) e impostando il progressivo di riga a -1 ("prog\_riga": [1,-1])

```

{
  "sigla": [
    [
      1,
      "OC"
    ]
  ],
  "serie": [
    [
      1,
      1
    ]
  ],
  "numero": [
    [
      1,
      330
    ]
  ],
  "id_riga": [
    [
      1,
      1
    ]
  ],
  "prog_riga": [
    [1,-1]
  ],
  "quantita": [
    [
      1,
      2
    ]
  ]
}
    
```

### 3.10. ENTITÀ UBICAZIONI

L'entità delle ubicazioni viene gestita in modo del tutto analogo alle altre risorse, ad eccezione di un particolare filtro che deve essere applicato in query string.

Per tutti i metodi (GET,POST,PUT,DELETE) l'entità *ubicazioni* è raggiungibile al seguente path

**[https://base\\_address/webapi/risorse/ubicazioni](https://base_address/webapi/risorse/ubicazioni)**

Per la GET della singola entità, oltre a specificare l'id (obbligatorio), è previsto anche un campo facoltativo per la gestione dei progressivi delle ubicazioni: *cod\_art\_progr*

Questo parametro, da inserire in query string, può assumere i seguenti valori:

- Nessun valore: vengono ritornati tutti i progressivi degli articoli in quella ubicazione
- Codice articolo: vengono ritornati tutti i progressivi dell'articolo specificato in quella ubicazione

Se il parametro non è presente vengono ritornati solo i campi dell'anagrafica dell'ubicazione senza i progressivi.

Esempio di lettura di tutti i progressivi di una ubicazione

```
GET https://base_address/webapi/risorse/ubicazioni/76?cod_art_progr
```

Esempio di lettura dei progressivi di un articolo in una ubicazione

```
GET https://base_address/webapi/risorse/ubicazioni/76?cod_art_progr=FUELEX97
```

### 3.11. TRASFORMAZIONE DOCUMENTI

Per tutte le risorse sotto l'entità documenti è possibile specificare in query string in fase di modifica (PUT) o creazione (POST) l'eventuale sigla documento in cui si vuole trasformare il documento senza doversi preoccupare di gestire la trasformazione "a mano" eliminando il documento di origine o gestendo i residui delle righe. Eventuali moduli di stampa dovranno essere specificati nel campo dedicato *cod\_modulo*.

Per le trasformazioni da documenti di tipo ordine cliente OC, se si vogliono gestire anche i residui di riga, è necessario valorizzare anche le seguenti nuove variabili di riga (array) a seconda che la riga sia gestita a quantità, taglia, lotto, colli.

```
"nr_colli_res": "Numero colli residuo"
"quantita_res": "Quantita' / Peso Lordo residuo"
"qta_taglia_res": "Quantita' per taglia residuo"
"qta_lotto_res": "Lotti - Quantita' residuo"
"nr_colli_lo_res": "Lotti - Colli residuo"
"qta_tg_lo_res": "Lotti - Quantita' per taglia residuo"
```

La differenza sostanziale tra l'utilizzo dei metodi POST e PUT è che il metodo PUT, per sua natura, non restituisce alcuna informazione mentre il metodo POST ritorna sempre l'ID della risorsa creata nel campo *Location* nell'header della response. Inoltre con il metodo POST è possibile indicare sempre anche righe provenienti da documenti diversi, con il metodo PUT invece si trasforma esclusivamente il singolo documento indicato.

#### PUT

È sufficiente specificare la sigla di destinazione in query string come nell'esempio che segue

```
PUT https://base_address/webapi/risorse/documenti/ordini-clienti/OC+1+1?sigla_trasformazione=BC
```

In questo esempio si trasforma l'ordine OC1/1 in un D.d.t. (BC)

Le quantità a residuo devono essere intese (come in emissione/revisione documenti) come le quantità che verranno evase e la differenza andrà a residuo. Ad esempio se la prima riga ha quantità 10, se viene valorizzata

```
"quantita_res": [
  [1,3]
]
```

si intende che, della prima riga con indice 1, vengono evase 3 quantità e 7 rimangono a residuo nel documento originale. Il documento di destinazione verrà creato con numerazione automatica.

Dovranno sempre essere indicati tutti gli indici delle righe che si vogliono trasformare e, nel caso di OC, impostare tutte le righe in evadibili "E".

Nell'esempio che segue è presente una sola riga, ed omettendo le quantità a residuo viene evasa completamente.

```

{
  "id_riga": [
    [
      1,
      1
    ]
  ],
  "tp_riga": [
    [
      1,
      "R"
    ]
  ],
  "tipo_stato_riga": [
    [
      1, "E"
    ]
  ]
}
    
```

## POST

In questo caso, dato che si sta creando un nuovo documento, dovranno essere specificati nel JSON con i dati di input anche tutti i documenti di origine.

**POST [https://base\\_address/webapi/risorse/documenti/movimenti-magazzino?sigla\\_trasformazione=BC](https://base_address/webapi/risorse/documenti/movimenti-magazzino?sigla_trasformazione=BC)**

Ad esempio se si vuole fare una fattura (FT) partendo da 2 bolle (BC) differenti è necessario valorizzare anche gli elementi "sigla\_doc\_orig", "serie\_doc\_orig", "numero\_doc\_orig", "data\_doc\_orig" indicando per ogni riga il suo riferimento di origine come nell'esempio che segue

**POST [https://base\\_address/webapi/risorse/documenti/movimenti-magazzino?sigla\\_trasformazione=FT](https://base_address/webapi/risorse/documenti/movimenti-magazzino?sigla_trasformazione=FT)**

```

{
  "sigla": "FT",
  "serie": 1,
  "numero": 0,
  "cod_conto": "501.00002",
  "id_riga": [
    [
      1,
      1
    ],
    [
      2,
      2
    ]
  ],
  "tp_riga": [
    [
      1,
      "R"
    ],
    [
      2,
      "R"
    ]
  ],
  "data_documento": "20220523",
  "codice_articolo": [
    [
      1,
      "ATT001"
    ],
    [
      2,
    ]
  ]
}
    
```

```

    "ATT003"
  ]
},
"quantita":[
  [
    1,
    3
  ],
  [
    2,
    4
  ]
],
"cod_iva":[
  [
    1,
    "23"
  ],
  [
    2,
    "23"
  ]
],
"sigla_doc_orig":[
  [1,"BC"],
  [2,"BC"]
],
"serie_doc_orig":[
  [1,1],
  [2,1]
],
"numero_doc_orig":[
  [1,2],
  [2,12]
],
"id_rif_testata":[
  [1,1],
  [2,2]
],
"data_doc_orig":[
  [1,"20220520"],
  [2,"20220523"]
]
}

```

- se nel body del POST è specificata una sigla differente viene ignorata, la query string *sigla\_trasformazione* è prioritaria.
- nel caso di trasformazione da Ordini (OC) a documento superiore è necessario specificare anche il *"tipo\_stato\_riga"* in evadibile "E".

### 3.12. WEBAPI PER LA RICHIESTA DI SERVIZI

In tutti i casi in cui l'operazione che si vuole effettuare sul gestionale non identifichi una risorsa ben precisa, ma piuttosto richieda l'esecuzione di una operazione o un insieme di operazioni che dovranno essere svolte da Mexal/Passcom, l'approccio RESTful non è più sufficiente e bisogna adottare un paradigma differente.

WebAPI mette a disposizione un ulteriore canale per la gestione di questi casi identificato con il nome *servizi*.

Le *http request* in questo caso saranno tutte di tipo *POST* e richiederanno l'esecuzione di un servizio.

Il path deve essere quindi formato in questo modo

**POST [https://base\\_address/webapi/servizi](https://base_address/webapi/servizi)**

Tutti i dati relativi all'esecuzione del particolare comando dovranno essere specificati nel *body* della *request*.

## COLLAGE SERVER REMOTO

Questo comando permette l'esecuzione di un Collage Server Remoto contenuto in una Passapp esattamente come avviene con WebShaker.

In questo caso è necessario specificare i dati necessari all'individuazione dell'etichetta Collage da eseguire all'interno della Passapp indicando *codice\_app*, *nome\_collage* e *etichetta\_collage*

Esempio di esecuzione di un Collage Server Remoto

Richiesta

**POST https://base\_address/webapi/servizi**

```
{
  "cmd": "esec_collage_server_remoto",
  "codice_app": "924148WEBSHAKER",
  "nome_collage": "colws",
  "etichetta_collage": "WAPI"
  "dati": {<eventuali dati da inviare>}
}
```

Risposta

```
{
  <risultato dell'elaborazione>
}
```

## SVILUPPO DISTINTA BASE

Servizio per lo sviluppo della distinta base di un articolo

Nel caso di articolo non a taglie è necessario specificare la quantità come primo elemento dell'array "quantita\_taglie".

Richiesta

**POST https://base\_address/webapi/servizi**

```
{
  "cmd": "sviluppo_distinta_base",
  "dati": {
    "codice_articolo": "<codice_articolo>",
    "codice_articolo_padre": "<codice_art_padre>",
    "codice_cliente": "<codice_cliente>",
    "id_riga_oc": 0,
    "quantita_taglie": [
      [1,n],
      ..
      [32,n]
    ]
  }
}
```

### 3.13. RIASSUNTO DEI COMANDI RESTFUL

	Metodo http	PATH	Query String	HEADER	BODY	Success http Code
<b>Creazione</b>	POST	https://base_address/webapi/<nome_risorsa>	NO	Authorization Content-Type Coordinate- Gestionali	{ <dati> }	201 (Created)
<b>Lettura</b>	GET	https://base_address/webapi/<nome_risorsa> https://base_address/webapi/<nome_risorsa>/ <codice>	fields max next	Authorization Content-Type Coordinate- Gestionali	{}	200 (OK)
<b>Aggiornamento</b>	PUT	https://base_address/webapi/<nome_risorsa>/ <codice>	NO	Authorization Content-Type Coordinate- Gestionali	{ <dati> }	204 (No Content)
<b>Eliminazione</b>	DELETE	https://base_address/webapi/<nome_risorsa>/ <codice>	NO	Authorization Content-Type Coordinate- Gestionali	{}	204 (No Content)
<b>Ricerca</b>	POST	https://base_address/webapi/<nome_risorsa>	fields max next	Authorization Content-Type Coordinate- Gestionali	filtri:[ {...}, {...} ]	200 (OK)

## 4. ACCESSO AI DATI DEL GESTIONALE

Per gli archivi ad un solo livello (articoli, clienti, fornitori etc.) per recuperare le informazioni di tutto l'archivio è sufficiente utilizzare qualsiasi comando REST con un'unica richiesta.

Invece per quanto riguarda archivi che presentano più livelli (tipicamente i documenti) può essere necessario effettuare più chiamate per avere i dati nella loro completezza.

Prendiamo come esempio i documenti: questi archivi sono composti da una tabella master e una tabella di dettaglio (testata + righe). Per avere quindi il dettaglio completo delle testate e di tutte le righe che compongono il documento sarà necessario utilizzare più chiamate REST.

Per i documenti c'è un contenitore dedicato in cui poi andranno specificate le varie entità a cui si vuole accedere.

Il path sarà di questo tipo

**COMANDO `https://base_address/webapi/risorse/documenti/<risorsa>`**

Prendiamo come esempio i documenti di tipo *ordini clienti*.

Per referenziare i dati di testata il path sarà di questo tipo

**GET `https://base_address/webapi/risorse/documenti/ordini-clienti`**

Con questo comando viene restituita una lista di tutte le testate dei documenti.

Se invece si vogliono avere i dettagli sulle righe dei documenti bisogna utilizzare un comando di questo tipo

**GET `https://base_address/webapi/risorse/documenti/ordini-clienti/righe`**

In questo caso verranno restituite tutte le righe di tutti i documenti di tipo *ordini-clienti*. Sarà poi compito del programmatore utilizzare i filtri in modo da leggere/scrivere solo le righe di determinati documenti.

Nel caso si debba specificare una chiave, bisogna comporla inserendo la sigla, la serie e il numero del documento in questo modo (per i movimenti di magazzino va aggiunto anche il codice del conto)

**GET `https://base_address/webapi/risorse/documenti/ordini-clienti/OC+1+55`**

**Nota: i campi chiave necessari possono essere recuperati visionando la risposta dell'HELP. Per ogni entità sarà presente un elemento "chiavi" contenente tutte le chiavi necessarie per completare la richiesta.**

**Nota: i dati restituiti da una GET su una lista possono essere leggermente differenti da quelli restituiti dalla GET di una singola entità. Ad esempio nelle liste gli array sono tutti al massimo di 2 livelli, mentre nelle singole entità gli array possono essere restituiti fino a 3 livelli.**

### 4.1 ESEMPIO GESTIONE ORDINI CLIENTI

Creazione di un ordine

Richiesta

**POST `https://base_address/webapi/risorse/documenti/ordini-clienti`**

```
{
  "id_riga": [
    [
      1,
      1
    ]
  ],
  "tp_riga": [
    [
      1,
```

```

    "R"
  ]
],
"mmart":[
  [
    1,
    "ATT002"
  ]
],
"mmali":[
  [
    1,
    "22"
  ]
],
"mmqta":[
  [
    1,
    3
  ]
],
"codice_agente":"601.00001",
"serie":1,
"sigla":"OC",
"numero":0,
"mmdat":"20210825",
"mmcli":"501.00022"
}

```

### Modifica di un ordine

Richiesta

**PUT** [https://base\\_address/webapi/risorse/documenti/ordini-clienti/OC+1+2](https://base_address/webapi/risorse/documenti/ordini-clienti/OC+1+2)

```

{
  "id_riga":[
    [
      1,
      1
    ]
  ],
  "tp_riga":[
    [
      1,
      "R"
    ]
  ],
  "codice_agente":"601.00002"
}

```

**NOTA: i dati che referenziano le righe del documento devono essere sempre tutti specificati. Le righe non dichiarate verranno eliminate. In questo esempio è stato indicato che nel documento esiste solo una riga e di tipo "R". Se il documento avesse avuto altre righe sarebbero state eliminate. ATTENZIONE!**

In questo caso si vuole modificare solo il codice agente lasciando tutto il resto inalterato. Gli indici di riga e il tipo di riga devono essere sempre specificati.

### Lettura di tutti gli ordini

Richiesta

**GET** [https://base\\_address/webapi/risorse/documenti/ordini-clienti](https://base_address/webapi/risorse/documenti/ordini-clienti)

### Lettura di un ordine

Richiesta

**GET** [https://base\\_address/webapi/risorse/documenti/ordini-clienti/OC+1+2](https://base_address/webapi/risorse/documenti/ordini-clienti/OC+1+2)

### Cancellazione di un ordine

Richiesta

**DELETE** [https://base\\_address/webapi/risorse/documenti/ordini-clienti/OC+1+2](https://base_address/webapi/risorse/documenti/ordini-clienti/OC+1+2)

### Lettura di tutte le righe di un particolare ordine

Richiesta

**POST** [https://base\\_address/webapi/risorse/documenti/ordini-clienti/righe/ricerca](https://base_address/webapi/risorse/documenti/ordini-clienti/righe/ricerca)

```
{
  "filtri":[
    {
      "campo":"sigla",
      "condizione":"=",
      "valore":"OC"
    },
    {
      "campo":"serie",
      "condizione":"=",
      "valore":1
    },
    {
      "campo":"numero",
      "condizione":"=",
      "valore":3
    }
  ]
}
```

Ritorna tutte le righe documento dell'OC1/3

**Nota:** le righe del documento, essendo una lista, possono essere modificate solo quando si esegue la PUT sul documento che le contiene specificando riga per riga i campi da modificare.

### Trasformazione di un ordine in d.d.t

**PUT** [https://base\\_address/webapi/risorse/documenti/ordini-clienti/OC+1+2?sigla\\_trasformazione=BC](https://base_address/webapi/risorse/documenti/ordini-clienti/OC+1+2?sigla_trasformazione=BC)

```
{
  "id_riga":[
    [
      1,
      1
    ]
  ],
  "tp_riga":[
    [
      1,
      "R"
    ]
  ]
}
```

Elimina il documento OC1/2 e crea la BC corrispondente

## 5. CHANGELOG

Di seguito saranno indicati tutti i cambiamenti di questo manuale rispetto alle versioni precedenti.

### V1.6: DALLA VERSIONE GESTIONALE 2022G – V824

#### Correzione anomalie

Esportato il campo "descr\_art\_ext" anche per la risorsa *movimenti-magazzino/righe* (Sol.66278)

#### Nuove funzionalità

Trasformazione documenti (vedi [sezione dedicata](#))

#### Nuove variabili

"nr\_colli\_res": "Numero colli residuo"  
 "quantita\_res": "Quantita' / Peso Lordo residuo"  
 "qta\_taglia\_res": "Quantita' per taglia residuo"  
 "qta\_lotto\_res": "Lotti - Quantita' residuo"  
 "nr\_colli\_lo\_res": "Lotti - Colli residuo"  
 "qta\_tg\_lo\_res": "Lotti - Quantita' per taglia residuo"

#### Nuove risorse

- Liste-prelievo (vedi [sezione dedicata](#))
- Giacenze articoli: nella risorsa articoli è stata aggiunta l'entità progressivi. Facendo una GET su questa entità verranno restituite tutte le giacenze dell'articolo selezionato. *Es. GET base\_address/articoli/COD\_ART/progressivi*  
*Verranno restituiti array di n record, e per ogni record è indicata la quantità di giacenza (inventario, carico, scarico) il magazzino, l'eventuale ubicazione.*
- *Alias-articoli: https://base\_address/webapi/risorse/alias-articoli (solo lista e ricerca)*
- *Imballi: https://base\_address/webapi/risorse/dati-general/imballi*

### V1.5.2: DALLA VERSIONE GESTIONALE 2022D – V821

#### Nuovi campi

- Nella risorsa *dati-general/particolarita* è stato aggiunto il campo **descr\_art\_cli**: *Descrizione articolo del Cli/For se "tipo\_part" = 'A'*
- Nella risorsa *ubicazioni* è stato aggiunto il campo **percorso**: *Codice percorso utilizzato per creare un percorso all'interno del magazzino*
- Nella risorsa *documenti/movimenti-magazzino* sono stati aggiunti i seguenti campi :

#### Campi di testata

```
{
    "nome": "rif_nr_bl",
    "descrizione": "Numero di riferimento bolla di lavoro",
    "tipo": "Numerico"
},
{
    "nome": "rif_nr_sbl",
    "descrizione": "Numero di riferimento sottobolla di lavoro",
    "tipo": "Numerico"
},
{
    "nome": "rif_nr_fs_lav",
    "descrizione": "Numero di riferimento fase di lavorazione",
```

```

        "tipo": "Numerico"
    }
    Campi di riga (array)
    {
        "nome": "cod_rif_pf_bl",
        "descrizione": "riferimento PF della BL; solo produzione",
        "tipo": "Numerico",
        "dimensione_array": "dinamico"
    },
    {
        "nome": "dt_doc_bl",
        "descrizione": "data documento della BL; solo produzione",
        "tipo": "Alfanumerico",
        "dimensione_array": "dinamico"
    },
    {
        "nome": "nr_bl",
        "descrizione": "numero BL; solo produzione",
        "tipo": "Numerico",
        "dimensione_array": "dinamico"
    },
    {
        "nome": "nr_sbl",
        "descrizione": "sotto BL; solo produzione",
        "tipo": "Numerico",
        "dimensione_array": "dinamico"
    },
    {
        "nome": "prog_fase_bl",
        "descrizione": "fase BL; solo produzione",
        "tipo": "Numerico",
        "dimensione_array": "dinamico"
    }
    }
    
```

#### Correzione refusi

- Nelle risorse *documenti/ordini documenti/preventivi documenti/matrici* il campo *id\_righe\_dba* è stato cambiato in *id\_rec\_dba\_dbvo* ma non documentato
- Il Content-type è opzionale esclusivamente per in metodi GET e DELETE per i metodi POST e PUT è obbligatorio

## V1.5.1: DALLA VERSIONE GESTIONALE 2022C2 – V820B

#### Correzione refusi

Nel servizio *sviluppo\_distinta\_base* sono stati corretti i nomi di 2 variabili:

*escludi\_da\_deposito* (mancava la **I**)

*escludi\_da\_scarico* (mancava la **I**)

Nell'entità *dati-general/particolarita* è stato corretto il nome di una variabile:

erano stati erroneamente dichiarati due campi con lo stesso nome (`cod_articolo`), è stato lasciato `cod_articolo` per indicare il codice dell'articolo mentre è stato aggiunto `cod_art_cli` per indicare il codice articolo associato al cliente/fornitore

## V1.5: DALLA VERSIONE GESTIONALE 2022C1 – V820A

### Nuove risorse

Archivio Lotti: aggiunti relativi riferimenti all'id del lotto anche nelle risorse che lo referenziano

Archivio Ubicazioni ([vedi sezione dedicata](#)): aggiunti relativi riferimenti all'id dell'ubicazione anche nelle risorse che la referenziano

### Nuovi campi

Aggiunto campo `tp_articolo` (tipologia articolo) nella risorsa degli impegni e nel servizio per lo sviluppo della distinta base `sviluppo_distinta_base`

## V1.4: DALLA VERSIONE GESTIONALE 2022B4 – V819D

### Nuove variabili

Nella risorsa `articoli` (sia lista che singola entità) sono state aggiunte le seguenti variabili che indicano la presenza o meno di immagini (S/N):

"img\_art\_disp": indica se l'immagine articolo è presente

"img\_cat\_disp": indica se l'immagine di catalogo è presente

"img\_icona\_disp": indica se l'icona è presente

## V1.3: DALLA VERSIONE GESTIONALE 2022B – V819

Correzione refusi di tipi di dato tra liste e singole risorse:

A=alfanumerico

F=float

N=numerico

### Clients / Fornitori

"gest\_rit\_acc" **Impostato tipo 'A' S/N in get/put**

"alq\_regio\_sport" **Impostato tipo 'F' in lista**

"alq\_comun\_sport" **Impostato tipo 'F' in lista**

"fido\_migliaia" **In lista rimane solo "fido" mentre in singola entità "fido" e "fido\_migliaia"**

"anag\_dt\_mod" **Impostato tipo 'A' in lista**

### Articoli

"larghezza\_ubic" **Impostato tipo 'F' in get/put**

"altezza\_ubic" **Impostato tipo 'F' in get/put**

"profondita\_ubic" **Impostato tipo 'F' in get/put**

"peso\_ubicazione" **Impostato tipo 'F' in get/put**

"cod\_srv\_intra" **Impostato tipo 'A' numero formattato con '0' a sx in get/put come in lista**

### Ordini / Preventivi / Matrici

"tp\_cambio\_vlt" **Impostato tipo 'A' S/N in get/put**

"tp\_cambio\_euro" **Impostato tipo 'A' S/N in get/put**

"gest\_mydb\_testa" **Impostato tipo 'A' S/N in get/put**

"gest\_mydb\_riga" **Impostato tipo 'A' S/N in get/put**

### Movimenti di magazzino

"tp\_cambio\_vlt" **Impostato tipo 'A' S/N in get/put**

"tp\_cambio\_euro" **Impostato tipo 'A' S/N in get/put**

"gest\_mydb\_testa" **Impostato tipo 'A' S/N in get/put**

"tipo\_stato\_riga" **Impostato tipo 'A' in lista**

"gest\_mydb\_riga" **Impostato tipo 'A' S/N in get/put**

## V1.2: DALLA VERSIONE GESTIONALE 2022A – V818

### Correzione refusi:

- [Sviluppo distinta base](#): quantità obbligatoria

### Modifiche al protocollo:

- Nuove risorse:
  - Fasi distinte base
  - [Archivi MyDB](#)
  - Aggiunta la risorsa "conti" oltre ai clienti e fornitori
- Nuova nomenclatura delle variabili: per tutti i dettagli usare il [comando HELP e info=true](#). Per la conversione tra vecchia e nuova nomenclatura vedere [Allegato A](#)
- Corretta sintassi filtri per gestire [array a due livelli](#)
- Aggiunti i campi *id\_riga\_orig* e *cat\_annul\_riga* per i documenti di magazzino e degli ordini

## V1.1 : DALLA VERSIONE GESTIONALE 2021I – V816

### Modifiche al protocollo:

- Aggiunto il parametro opzionale Magazzino nell'header della richiesta per permettere di specificare il magazzino di default
- Le liste sono state arricchite di ulteriori dati e gli array possono arrivare fino a 2 livelli invece che 1 solo livello
- Cambiato path del [comando HELP](#) e aggiunto filtro *extended* in query string
- Aggiunte nuove risorse: usare il [comando HELP](#) per tutti i dettagli
- Cambiato il path di alcune risorse: usare il [comando HELP](#) per tutti i dettagli
- Aggiunto [limite al numero massimo di utenti](#) appartenenti al gruppo WebAPI: 3
- Aggiunto servizio per lo [sviluppo della distinta base](#)
- Eliminata documentazione sull'accesso alle risorse tramite il canale *servizi*: si consiglia di utilizzare sempre l'accesso con logica REST sul path "risorse"
- Inserita [entità allegati](#): in questa versione solo lettura immagini articoli

- Non è più necessario gestire il cookie [per installazioni Live](#)
- Nell'entità di tipo ricerca aggiunta [la condizione "contiene"](#)

**Nota:**

- nei prossimi rilasci di WebAPI i nomi delle chiavi dei JSON verranno cambiati gestendo un massimo di 15 caratteri. In questo modo i nomi degli elementi risulteranno più parlanti e più aderenti ad una logica RESTful. Nel caso siano già state fatte applicazioni con la nomenclatura attuale, di volta in volta verranno rilasciate delle tabelle di correlazione tra i vecchi ed i nuovi nomi dei vari archivi aggiornati. In ogni caso per i dettagli sui nuovi nomi dei campi si può usare il metodo GET con il parametro *info=true* in query string.

**Correzione refusi:**

- `esec_collage_server_remoto`: era stato erroneamente indicato `esegui_collage_server_remoto`

## 6. ALLEGATI

### ALLEGATO A

#### Nuova nomenclatura

Tutti i nomi dei campi sono stati cambiati per essere più "parlanti" ed aderenti alle linee guida REST. I nomi sono tutti minuscoli, con parole separate da underscore "\_" e di lunghezza massima di 15 caratteri.

Oltre ai nomi sono state apportate alcune modifiche anche alla struttura di alcuni archivi nell'ottica di facilitare l'accesso alle risorse. Di seguito sono elencate le modifiche più importanti:

- Correlazione tra vecchio nome campo (1^ colonna) e nuovo nome campo (2^ colonna)
- Alcuni campi sono stati eliminati e quindi non è presente il vecchio nome in questa lista
- Alcuni campi sono stati aggiunti. Il vecchio nome campo (1^ colonna) non è presente
- Alcuni campi array sono stati uniti; ad esempio array taglie da 1 a 16 e array taglie da 17 a 32: nella terza colonna è indicato: "nomevecchio1" + "nomevecchio2" (es. taglie articolo -> array da 32 elementi)
- Alcuni campi flag che ritornavano 1/0 ora ritornano S/N. Nella terza colonna è indicato: Vale S/N invece di 1/0
- Alcune variabili array contenenti un flag per ogni elemento sono state trasformate in variabili singole: nel vecchio nome campo (1^ colonna) viene indicato l'elemento dell'array: "nomevecchio(4)" (es. "Condizioni destinatario" nel caso di "Fatturare a")

#### Tabella Gruppi e Mastri

"cmgru"	"id_gruppo"
"cmmas"	"mastro"
"cmdes"	"descrizione"
"cmtpg"	"tp_gruppo"
"cmtpm"	"tipo"
"cmtpb"	"tp_bilancio"
"cmtpa"	"mastro_banca"
"cmtps"	"mastro_sospeso"
"cmcda"	"bil_cee_ecc_d"
"cmcav"	"bil_cee_ecc_a"
"cmucd"	"bil_ue_ecc_d"
"cmuca"	"bil_ue_ecc_a"
"cmrcd"	"bil_rsm_ecc_d"
"cmrca"	"bil_rsm_ecc_a"
"cmecd"	"bil_ets_ecc_d"
"cmeca"	"bil_ets_ecc_a"
"cmcat"	"categoria"

#### Tabella Valute

"mudiv"	"id"
"musig"	"sigla"
"mudes"	"descrizione"
"mudec"	"nr_dec_importi"
"murap"	"cambio_euro"
"muord"	"sequenza_visual"
"mutip"	"tipo"
"muide"	"simbolo"
"muprz"	"nr_dec_prezzi"

## Tabella Esenzioni Iva

"ceali"	"codice"
"cedes"	"descrizione"
"cepla"	"plafond"
"cedse"	"descr_estesa"
"cetpo"	"tp_operazione"
"cecde"	"id_descrizione"
"cecci"	"id_classifica"
"cenpa"	"codice_natura"

## Tabella Pagamenti

"cpnum"	"id"
"cpdes"	"descrizione"
"cptis"	"tp_sconto"
"cpsco"	"sconto_perc"
"cpspi"	"gest_spese_inc"
"cpspb"	"gest_spese_bol"
"cpsbf"	"gest_inizio_sca"
"cpscf"	"scadenza_fissa"
"cpscg"	"giorni_proroga"
"cprap"	"gest_rate_perc"
"cpdst"	"descr_pnota"
"cpptd"	"paga_tot_doc"
"cpccr"	"cos_ric_pg_ab"
"cpcpg"	"causale_pg"
"cpcab"	"causale_ab"
"cpabp"	"conto_ab_pas"
"cpaba"	"conto_ab_att"
"cptip"	"tp_pagamento"
"cpsca"	"giorni_scadenza"
"cptpr"	"tp_rata_iva"
"cpper"	"percentuale"

## Archivi Clienti e Fornitori

"pccod"	"codice"
"pccal"	"cod_alternativo"
"pcidrubuni"	"id_rubrica_uni"
"pcnaz"	"tp_nazionalita"
"pcpae"	"cod_paese"
"pcpri"	"gest_privato"
"pcpfs"	"gest_per_fisica"
"pccfi"	"codice_fiscale"
"pcnpi"	"partita_iva"
"pcnpiext"	"ext_piva_estera"
"pcpis"	"piva_secondaria"
"pcnopiva"	"gest_no_piva"
"pcdenom"	"denominazione"
"pcdes"	"ragione_sociale"
"pcdad"	"dt_valid_ragsoc"
"pcnad"	"ragsoc_prec"
"pccog"	"cognome"
"pcnom"	"nome"
"pcdtn"	"dt_nasc_cost"
"pclon"	"loc_nasc_cost"
"pcprn"	"prv_nasc_cost"
"pcind"	"indirizzo"
"pccap"	"cap"

"pcloc"	"localita"
"pcpro"	"provincia"
"pctel"	"telefono"
"pcfax"	"fax"
"pcint"	"email"
"pcint1"	"url"
"pcpec"	"pec"
"pcsportivo"	"gest_sportivo"
"pcsportazi"	"gest_sport_escl"
"pcsportalr"	"alq_regio_sport"
"pcsportalc"	"alq_comun_sport"
"pcpaga"	"cod_pg_no_deper"
"pcpagd"	"cod_pg_deper"
"pcalc"	"alq_comp"
"pcric"	"nome_ricerca"
"pctpsoc"	"tp_societa"
"pclotte"	"cod_lotteria"
"pcentep"	"gest_ente_pa"
"pcsplitpg"	"gest_spl_pay_pa"
"pctpfatte"	"gest_fatt_el_pa"
"pcfattele"	"gest_fatt_el"
"pcdfattel"	"dt_gest_fatt_el"
"pcseriele"	"serie_fatt_el"
"pcallxml"	"allega_fatt_el"
"pcrallfrm"	"cod_modu_allega"
"pcrifammpa"	"rif_ammin_pa"
"pccodupa"	"cod_ufficio_pa"
"pcobrfampa"	"gest_rif_obbl"
"pccodsdi"	"codice_sdi"
"pcnospesom"	"gest_spesometro"
"pcopturis"	"gest_spes_turis"
"pcdocriep"	"gest_riep_spes"
"pcesterom"	"trattam_esterom"
"pcagri34c6"	"gest_agricolo"
"pcdomu"	"gest_condominio"
"pcselfbill"	"gest_self_bill"
"pcrgfisc"	"reg_fis_fatt_el"
"pcivadiff"	"gest_iva_differ"
"pcnba"	"id_banca"
"pcabi"	"cod_abi"
"pccab"	"cod_cab"
"pcccb"	"conto_corren"
"pcbbcin"	"cin_bban"
"pcibpae"	"pae_iban"
"pcibcin"	"cin_iban"
"pcibcod"	"iban_libero"
"pcbban"	"bban"
"pciban"	"iban"
"pcbic"	"bic"
"pcasi"	"assog_iva"
"pcval"	"valuta"
"pcopcharg"	"tp_op_reverse"
"pccst"	"cod_cat_sta"
"pcona"	"cod_zona"
"pcbcs"	"gest_add_bollo"
"pcpag"	"cod_pagamento"
"pcscf"	"scadenza_fissa"
"pcspc"	"giorni_proroga"
"pcmes"	"mesi_esclusi"
"pcmap"	"gg_anticipo"
"pcref"	"gest_rg_eff"
"pccob"	"cod_banca"
"pccigcup"	"gest_cig_cup"
"pcbap"	"banca_appoggio"
"pcsdmdnd"	"nr_mandato_rid"

"pccdrd"	"cod_rid_cliente"
"pcridtpi"	"tp_incasso"
"pcriddtm"	"dt_mandato_rid"
"pcridsts"	"stato_sequenza"
"pcubbp"	"banca_appo_agg"
"pcubsddmnd"	"nr_mandato_agg"
"pcubbic"	"bic_agg"
"pcubpae"	"pae_iban_agg"
"pcubibcin"	"cin_iban_agg"
"pcubibcod"	"iban_agg"
"pcubbacin"	"cin_bban_agg"
"pcubabi"	"abi_bban_agg"
"pcubcab"	"cab_bban_agg"
"pcubccb"	"conto_bban_agg"
"pcubiban"	"iban_codif_agg"
"pcubbban"	"bban_codif_agg"
"pcubcodrid"	"cod_rid_cli_agg"
"pcsddcodi"	"nr_mandato_sepa"
"pcubriddtm"	"dt_mandato_cli"
"pcsdddatma"	"dt_mandato_sepa"
"pcubridsts"	"stato_sepa_cli"
"pcsddstat"	"stato_seq_sepa"
"pcubridtpi"	"tp_incasso_sepa"
"pcsddtipo"	"tp_mandato_sepa"
"pcrsp"	"responsabile"
"pcsp"	"anticipo_spese"
"pcazi"	"sigla_azienda"
"pccpf"	"gest_cont_parc"
"pcstu"	"cod_for_studio"
"pcost"	"cod_costo_prest"
"pccpg"	"gest_registr_pg"
"pcppg"	"cod_conto_pg"
"pcuteid"	"id_utente_parc"
"pctpra"	"tp_modello_redd"
"pccpra"	"cod_prat_redd"
"pclin"	"cod_lingua"
"pclis"	"cod_listino"
"pclie"	"gest_extra_lis"
"pccsc"	"cod_cat_sconti"
"pcsc"	"sconto_icz"
"pcage"	"cod_agente"
"pcagecond"	"cod_cond_age"
"pccpr"	"cod_cat_pr"
"pcrbl"	"gest_rg_doc"
"pciev"	"gest_ordine_e"
"pcfta"	"cod_fatturare_a"
"pcspediz"	"cod_ind_sped"
"pctrm"	"tp_trasporto"
"pcvet"	"cod_vettore"
"pcvpo"	"tp_porto"
"pcvde"	"descr_porto"
"pcmoc"	"gest_colli_auto"
"pcmop"	"gest_peso_auto"
"pcvsp"	"tp_spese_sped"
"pcvva"	"spese_sped"
"pcaeb"	"cod_asp_beni"
"pcincot"	"cod_incoterm"
"pcniscalbo"	"num_iscr_albo"
"pcdiscalbo"	"dt_iscr_albo"
"pcpardest(1)"	"gest_dst_lis_ve"
"pcpardest(2)"	"gest_dst_cat_sc"
"pcpardest(3)"	"gest_dst_sc_icz"
"pcpardest(4)"	"gest_dst_cat_pr"
"pcpardest(5)"	"gest_dst_part"
"pcpardest(6)"	"gest_dst_agente"

"pcpardest(7)"	"gest_dst_lingua"
"pcpardest(8)"	"gest_dst_vett"
"pcpardest(9)"	"gest_dst_porto"
"pcpardest(10)"	"gest_dst_spese"
"pcupardest(1)"	"gest_dst_mezzo"
"pcupardest(2)"	"gest_dst_colli"
"pcupardest(3)"	"gest_dst_peso"
"pcupardest(4)"	"gest_dst_beni"
"pcupardest(5)"	"gest_dst_agg"
"pcvac"	"gest_val_cauz"
"pcimb"	"qta_cauzioni"
"pcctrppc"	"cod_conto_ric"
"pcctrcon"	"cod_conto_uric"
"pcctrpma"	"cod_conto_merce"
"pcctrpml"	"cod_clav"
"pcctrpmp"	"cod_conto_cdep"
"pcctrpms"	"cod_conto_spesa"
"pcctrpmz"	"cod_conto_prest"
"pcsra"	"gest_rit_acc"
"pcalqrd"	"alq_rit_acc"
"pcqimprda"	"perc_impon_rit"
"pccauprda"	"cod_pg_770S"
"pcrlcogno"	"cognome_legale"
"pcrlnome"	"nome_legale"
"pcrlcdf"	"cod_fis_legale"
"pcrldtn"	"dt_nasc_legale"
"pcrllon"	"loc_nasc_legale"
"pcrlprn"	"prv_nasc_legale"
"pcrlpae"	"cod_pae_legale"
"pcanlcf"	"cod_analitico"
"pcriripcf"	"cod_crit_ripart"
"pctsaout"	"gest_no_tx_ts"
"pccatsplit"	"gest_spl_pay"
"pcdtspl"	"dt_spl_pay"
"pcctr"	"cod_tributo"
"pctpriten"	"tp_rit"
"pcmani"	"tp_regime_agv"
"pcdmani"	"dt_fine_regime"
"pcgsage"	"gest_regime_agv"
"pcfdp"	"perc_fondo_prof"
"pctpre"	"cod_prestazione"
"pccatstd"	"cod_catasto_dom"
"pcglad"	"gest_glad"
"pcalcol"	"perc_contr_glad"
"pcaldft"	"perc_ded_glad"
"pcatcol"	"cod_attiv_glad"
"pcenainira"	"dt_inizio_mndto"
"pcenafinra"	"dt_fine_mndto"
"pcenavarra"	"dt_variaz_mndto"
"pcenatpatt"	"tp_mndto"
"pcenaimptr"	"prog_impon_enas"
"pcenaimprt"	"rett_impon_enas"
"pcenamdfir"	"rett_impon_firr"
"pcenatpman"	"tp_mndto_corr"
"pcenacontr"	"contr_calc_enas"
"pcenadovut"	"contr_comp_enas"
"pcenainteg"	"integr_enas"
"pctprecap2"	"tp_recap"
"pcnome2"	"nome_recap"
"pcind2"	"ind_recap"
"pccap2"	"cap_recap"
"pcloc2"	"loc_recap"
"pcprov2"	"prv_recap"
"pcindeff"	"gest_recap_eff"
"pcrating"	"tp_rating"

"pcins"	"nr_insoluti"
"pcvin"	"tot_insoluti"
"pcfido"	"fido"
"pcfifid"	"fido_migliaia"
"pcfifidtpmsg"	"gest_fuori_fido"
"pcggrit"	"nr_gg_rit_pg"
"pcnds"	"descr_nota"
"pcnde"	"dst_nota"
"pcndt"	"dt_sca_nota"
"pcvdr"	"vid_anag_rip"
"pcvdf"	"vid_anag_no_rip"
"pcwww"	"vid_aggiuntive"
"pcdcr"	"anag_dt_crea"
"pcdum"	"anag_dt_mod"
"pcfmo"	"anag_mod"
"pcmra"	"stato_anag_att"
"pcmr"	"stato_anag_var"
"pcdtsist"	"anag_dt_scritt"
"pcorsist"	"anag_ora_scritt"
"pcann"	"conto_precanc"
"utenteum"	"utente_ult_mod"
"dataum"	"data_ult_mod"

## Tabella Descrizioni Magazzini

"manum"	"id"
"mades"	"descrizione"
"maval"	"gest_valorizza"
"manot"	"nota"
"macap"	"cap"
"maloc"	"localita"
"mapro"	"provincia"
"maind"	"indirizzo"
"maann"	"annullato"

## Tabella Sconti Listini

"mscli"	"cod_cat_cli"
"msart"	"cod_cat_art"
"mssco"	"sconto"
"msscn"	"sconto_esteso"

## Tabella Provvigioni Listini

"mpcli"	"cod_cat_cli"
"mpart"	"cod_cat_art"
"mppro"	"provvigione"
"mptip"	"tipo"
"mpfor"	"formula"
"mpage"	"cod_agente"
"mpana"	"cod_cond_agente"

## Tabella Listini

"mInum"	"id"
"mlart"	"cod_cat_art"
"mldes"	"descr_corta"
"mldse"	"descr_estesa"
"mldsc"	"descrizione"
"mlnot"	"nota"
"mlval"	"valuta"
"mldat"	"data_cambio"
"mldag"	"giorno_cambio"
"mldam"	"mese_cambio"
"mldaa"	"anno_cambio"
"miiva"	"gest_con_iva"
"mlarr"	"tp_arrot"
"mlimp"	"imp_fino_a"
"mlali"	"arrot_a"
"mlolt"	"arrot_oltre"
"mlprz"	"tp_calcolo"
"mlcpr"	"perc_ricarico"

## Tabella Taglie

"mtser"	"cod_serie"
"mtnum"	"numero_taglia"
"mtdss"	"descr_serie"
"mtmax"	"num_massimo_tg"
"mtdes"	"descr_taglia"
"mtcoe"	"coeff_consumo"

## Tabella Serie Documenti

"mfsig"	"sigla_documento"
"mfsmx"	"serie_massima"
"mfbcsc"	"descrizione"

## Tabella Sconti Quantità

"mqart"	"cod_cat_art"
"mqnum"	"prog_sconto"
"mqcli"	"cod_cat_cli"
"mqqta"	"fino_a"
"mqscoc"	"sconto"
"mqscnc"	"sconto_esteso"

## Tabella Omaggi/Abbuoni/Garanzie

"gscod"	"codice"
"gsdes"	"descrizione"
"gsded"	"gest_esenz_iva"
"gsdee"	"descr_estesa"

## Tabella Strutture Articoli

"mrnum"	"codice"
"mrdes"	"descrizione"
"mraut"	"autocod_da_pos"
"mrmrat"	"gest_automatica"
"mrmal"	"compos_cod_alt"
"mrmde"	"compos_desc"
"mralt"	"gest_cod_alt"
"mrdbp"	"posizione_dbp"
"mrcar"	"posizione_cart"
"mrprg"	"prog_elemento"
"mrcre"	"gest_creazione"
"mrazz"	"gest_azz_costi"
"mresc"	"gest_no_padre"
"mrobb"	"gest_qta_obbl"
"mrribb"	"gest_inter_obbl"
"mrrpv"	"gest_videate"
"mrcds"	"descrizione"
"mrctp"	"tp_elemento"
"mrcpo"	"pos_finale"
"mrnst"	"cod_struttura"
"mrqta"	"tp_tabellare"
"mrmds"	"descr_um"
"mrmra"	"coeff_conv"
"mrmda"	"pos_inizio_um"
"mrmap"	"pos_finale_um"
"mrusx"	"variabile_x"
"mrusy"	"variabile_y"
"mrusz"	"variabile_z"
"mrari"	"tp_arrot"

## Tabella Abbinamenti Colori

"trcod"	"codice"
"trnab"	"nr_abbinamenti"
"trabb"	"abbinamento"

## Tabella Categorie Statistiche Articoli

"myalf"	"sigla"
"mynum"	"numero"
"mydes"	"descrizione"

## Tabella Nature Articoli

"ggnat"	"codice"
"gnsta"	"stato"
"gndes"	"descrizione"
"gndtc"	"data_creazione"
"gndtm"	"data_modifica"

## Tabella Gruppi Merceologici

"gmmer"	"codice"
"gmsta"	"stato"
"gmdes"	"descrizione"
"gmdtc"	"data_creazione"
"gmdtm"	"data_modifica"
"gmpad"	"cod_grp_merc"
"gmumi"	"um"
"gmic"	"cod_ricavo"
"gmcos"	"cod_costo"
"gmweb"	"codice_web"

## Tabella Causali Movimenti

"mcnum"	"id"
"mcdes"	"descrizione"
"mccos"	"gest_cos_ultimo"
"mcdat"	"gest_dt_carico"
"mcval"	"gest_mov_valore"

## Articoli

"arcod"	"codice"	
"arnra"	"nr_record"	
"artip"	"tipologia"	
"arsme"	"tp_merce_rsm_fe"	
"arsmr"	"tp_merce_rsm_fr"	
"arstr"	"cod_struttura"	
"ardes"	"descrizione"	
"ardse"	"descr_estesa"	
"aragg"	"descrizione_agg"	
"aralt"	"cod_alternativo"	
"ariva"	"alq_iva"	
"arum1"	"um_principale"	
"arum2"	"um_secondaria"	
"ardec"	"decimali_qta"	
"artco"	"tp_confezione"	
"arcon"	"qta_collo"	
"arkoe"	"coeff_conv"	
"arkes"	"coeff_conv_ext"	
"arpta"	"peso_tara"	
"arvuc"	"cos_ult_valuta"	
"arcuc"	"cos_ult"	
"arvst"	"cos_std_valuta"	
"arsst"	"cos_std"	
"armft"	"qta_min_fatt"	
"armin"	"imp_min_fatt"	
"armax"	"imp_max_fatt"	
"arcps"	"ctrl_peso"	
"arqpr"	"qta_prendi"	
"arqpa"	"qta_paghi"	
"arimb"	"cod_imballo"	
"arsco"	"id_cat_sconto"	
"arscq"	"id_sconto_qta"	
"artpr"	"id_categoria_pr"	
"arlis"	"id_cat_prezzo"	
"arsta"	"sigla_cat_sta"	
"arstn"	"nr_cat_sta"	

"ardtc"	"dt_creazione"	
"ardta"	"dt_mod"	
"ardtas"	"dt_mod_sistema"	
"arric"	"cod_ricavo"	
"arcos"	"cod_costo"	
"arrso"	"cod_ric_sospeso"	
"artssc"	"cod_sp_sanit"	
"artst"	"tp_sp_sanit"	
"aranc"	"cod_analit_cos"	
"aranr"	"cod_analit_ric"	
"arpre"	"gest_precanc"	
"arann"	"gest annullato"	
"arpru"	"gest_preventivo"	
"arsos"	"cod_art_sost"	
"ardbp"	"gest_dbp"	
"ardbv"	"gest_dbv"	
"ardba"	"gest_dba"	
"arcar"	"gest_cart"	
"artmo"	"gest_tp_mdf"	
"arvmo"	"valore_mdf"	
"arsmo"	"gest_somma_mdf"	
"aramo"	"tp_arrot_mdf"	
"arro"	"val_arrot_mdf"	
"arrst"	"cod_struttura"	
"artde"	"tp_descrizione"	
"artpm"	"gest_tp_val_mdf"	
"arvup"	"valuta_cos_prec"	
"arcup"	"cos_prec"	
"arvul"	"valuta_cos_ult"	
"arcul"	"costo_ult"	
"arfor"	"cod_fornitore"	"arfor" + "arfore"
"arggr"	"gg_riordino"	"arggr" + "arggre"
"arlot"	"qta_riordino"	"arlot" + "arlote"
"arval"	"valuta_ri"	"arval" + "arvale"
"arfpr"	"prz_riordino"	"arfpr" + "arfpre"
"arcof"	"cod_riordino"	"arcof" + "arcofe"
"arqta"	"fino_qta_ri"	"arqta" + "arqtat"
"arscoq"	"sc_qta_ri"	"arscoq" + "arscqe"
"arscq"	"sc_qta_ri_ext"	"arscq" + "arscqe"
"arcma"	"cod_camp"	
"arcmu"	"um_camp"	
"arcmq"	"qta_camp"	
"arcm"	"qta_tg_camp"	"arcm" + "arcmxl"
"arcmp"	"tp_cond_camp"	
"arcmr"	"cod_art_camp"	
"arcmc"	"cod_cli_camp"	
"artxv"	"testo_varia"	
"artxt"	"testo_fisso"	
"arnds"	"nota"	
"arnde"	"gest_nota"	
"arndt"	"dt_sca_nota"	
"ardsl"	"descr_lingua"	
"arprz"	"prz_listino"	"arprz" + "arprze"
"arscoa"	"sconto_listino"	"arscoa" + "arscoe"
"arscon"	"sconto_lis_ext"	"arscon" + "arscone"
"arprotp"	"tp_pr_listino"	"arprotp" + "arprotpe"
"arproform"	"pr_formula"	"arproform" + "arproforme"
"arproagen"	"cod_agente"	"arproagen" + "arproagene"
"arprocond"	"id_cond_agente"	"arprocond" + "arproconde"
"arpro"	"provvigione"	"arpro" + "arproe"
"arper"	"perc_prz_lis"	"arper" + "arpere"
"arnumliscon"	"id_listino"	
"arvex"	"id_valuta_extra"	
"arexg"	"extra_fisso"	
"arexa"	"extra_a"	

"arexb"	"extra_b"	
"arexc"	"extra_c"	
"arctg"	"gest_cons_tg"	
"arfrm"	"gest_forma"	
"argrmer"	"cod_grp_merc"	
"arnatura"	"cod_natura"	
"arrdt"	"dt_gest_lotto"	
"arrtl"	"cod_tp_lotto"	
"arri"	"gest_rich_lotto"	
"arrsa"	"tp_sel_auto"	
"arenas"	"gest_enasarco"	
"arpr"	"nr_mesi_rateo"	
"argr"	"gest_ratei"	
"aragr"	"tp_alimento"	
"arubi"	"cod_ubicazione"	
"arsub"	"cod_sotto_ubic"	
"arcad"	"dt_ult_carico"	
"arsca"	"dt_ult_scarico"	
"arord"	"dt_ult_ordine"	
"ardtm"	"dt_ult_mov"	
"arsmi"	"scorta_massima"	
"arsma"	"scorta_minima"	
	"id_ubicazione"	
	"larghezza_ubic"	
	"altezza_ubic"	
	"profondita_ubic"	
	"peso_ubicazione"	
	"temp_min_ubic"	
	"temp_max_ubic"	
"arinq"	"qta_inventario"	
"arint"	"qta_inventa_tg"	"arint" + "arintxl"
"arinv"	"valore_inventa"	
"artin"	"tara_inventa"	
"arcin"	"colli_inventa"	
"arcarico"	"qta_carico"	
"arcat"	"qta_tg_carico"	"arcat" + "arcatxl"
"arcav"	"valore_carico"	
"artca"	"tara_carico"	
"arcca"	"colli_carico"	
"arqsc"	"qta_scarico"	
"arsct"	"qta_scarico_tg"	"arsct" + "arsctxl"
"arscv"	"valore_scarico"	
"artsc"	"tara_scarico"	
"arcsc"	"colli_scarico"	
"arorf"	"ord_fornitori"	
"arort"	"ord_for_taglie"	"arort" + "arortxl"
"arcli"	"ord_cli_e"	
"arcti"	"ord_cli_e_tg"	"arcti" + "arctixl"
"arcls"	"ord_cli_sps"	
"arcts"	"ord_cli_sps_tg"	"arcts" + "arctsxl"
"arcla"	"ord_cli_auto"	
"arcta"	"ord_cli_auto_tg"	"arcta" + "arctaxl"
"arprd"	"ord_produzione"	
"arctp"	"ord_prod_taglie"	"arctp" + "arctpxl"
"arino"	"cod_nomencl"	
"arima"	"gest_massa_net"	
"arim"	"coeff_massa_net"	
"arius"	"gest_unita_sup"	
"aridu"	"descr_unita_sup"	
"ariku"	"coeff_unita_sup"	
"aripr"	"cod_prv_intra"	
"aripa"	"cod_pae_intra"	
"arics"	"cod_srv_intra"	
"arivs"	"val_sta_unit"	
"armra"	"gest_attiva"	

"armrv"	"gest_variata"	
"arvu1"	"vid_art_no_rip"	
"arvu2"	"vid_art_rip"	
"arwww"	"vid_aggiuntive"	
"arpper"	"gest_period"	
"arpmes"	"mese_period"	
"arprit"	"gest_ritenuta"	
"arpcas"	"gest_cassa_pre"	
"argsi"	"gest_sep_inps"	
"arenpam"	"gest_enspam"	
"arenpals"	"gest_enspals"	
"arcatweb"	"cod_cat_web"	
"aroras"	"ora_mod_sistema"	
"utenteum"	"utente_ult_mod"	
"dataum"	"data_ult_mod"	

## Ordini Clienti, Ordini Fornitori, Preventivi, Matrici

"mmsig"	"sigla"	
"mmser"	"serie"	
"mmnum"	"numero"	
"mmcli"	"cod_conto"	
"mmdat"	"data_documento"	
"mmmodstp"	"cod_modulo"	
"mmcmo"	"id_causale"	
"mmmag"	"id_magazzino"	
"mmccr"	"id_costo_ricavo"	
"mmage"	"codice_agente"	
"mmagecond"	"id_cond_agente"	
"mmnot"	"nota"	
"mmmra"	"stato_doc_att"	
"mmrv"	"stato_doc_var"	
"mmidpassweb"	"id_sito_passweb"	
"mmest"	"id_valuta"	
"mmeve"	"cambio_vlt_euro"	
"mmdie"	"tp_cambio_vlt"	
"mmevc"	"cambio_euro_vlt"	
"mmdic"	"tp_cambio_euro"	
"mmmydbt"	"gest_mydb_testa"	
"mmdatcre"	"data_creazione"	
"mmoracre"	"ora_creazione"	
"mmutecre"	"utente_crea"	
"mmmencre"	"menu_creazione"	
"mmdatvar"	"dt_var_sistema"	
"mmoravar"	"ora_var_sistema"	
"mmutemod"	"utente_modifica"	
"mmmenmod"	"menu_modifica"	
"mmdocelett"	"gest_ord_el"	
"mmsigre"	"rif_sgl_attuale"	
"mmnumre"	"rif_nr_attuale"	
"mmdatre"	"rif_dt_attuale"	
"mmrifammpa"	"cod_rif_ammin"	
"mmtprifre"	"tp_riferimento"	
"mminps"	"gest_inps"	
"mmcapro"	"gest_previdenza"	
"mmnrcroriga"	"id_riga"	
"mmtpr"	"tp_riga"	
"mmart"	"codice_articolo"	
"mmtsd"	"tp_descrizione"	
"mmdee"	"descr_riga"	
"mmarr"	"cod_art_testo"	
"mmcau"	"qta_cauzione"	
"mmdes"	"descr_articolo"	
"mmdesagg"	"descr_art_ext"	

"mmcmr"	"id_causale_riga"	
"mmrmag"	"id_mag_riga"	
"mmtip"	"tipo_articolo"	
"mmcol"	"nr_colli"	
"mmqta"	"quantita"	
"mmkta"	"coeff_conv"	
"mmtag"	"qta_taglia"	"mmtag" + "mmtagxl"
"mmord"	"tipo_stato_riga"	
"mmprz"	"prezzo"	
"mmsco"	"sconto"	
"mmsca"	"dt_sca_riga"	
"mmpro"	"perc_provv"	
"mmali"	"cod_iva"	
"mmtagtab"	"cod_taglia"	
"mmpsi"	"rif_sgl_prev"	
"mmpse"	"rif_serie_prev"	
"mmpnu"	"rif_nr_prev"	
"mmpda"	"rif_dt_prev"	
"mmnrecriga"	"id_riga_dba"	
"mmmydbr"	"gest_mydb_riga"	
"mmrintrcrer"	"cod_rif_pf_bl"	
"mmrintrdat"	"dt_doc_bl"	
"mmrintrbnu"	"nr_bl"	
"mmrintrbsbl"	"nr_sbl"	
"mmrintrbfs"	"prog_fase_bl"	
"mmspe"	"id_ind_sped"	
"mmanspe"	"cod_anag_sped"	
"mmtsrf"	"tp_trasporto"	
"mmvet"	"cod_vettore"	
"mmpor"	"tp_porto"	
"mmdsp"	"descr_porto"	
"mmtps"	"tp_spese_sped"	
"mmvas"	"val_spese_sped"	
"mmaco"	"gest_colli_sped"	
"mmtco"	"nr_colli_sped"	
"mmape"	"gest_peso_sped"	
"mmtpe"	"peso_spedizione"	
"mmdtintr"	"dt_spedizione"	
"mmorintr"	"ora_spedizione"	
"mmpag"	"id_pagamento"	
"mmcbzpag"	"cod_banca"	
"mmacc"	"acconto_pagato"	
"mmspbanca"	"spese_banca"	
"mmspbolli"	"spese_bolli"	
"utenteum"	"utente_ult_mod"	
"dataum"	"data_ult_mod"	

## Movimenti di Magazzino

"mmsig"	"sigla"	
"mmser"	"serie"	
"mmnum"	"numero"	
"mmcli"	"cod_conto"	
"mmdat"	"data_documento"	
"mmmodstp"	"cod_modulo"	
"mmcmo"	"id_causale"	
"mmmag"	"id_magazzino"	
"mmccr"	"id_costo_ricavo"	
"mmage"	"codice_agente"	
"mmagecond"	"id_cond_agente"	
"mmnot"	"nota"	
"mmmra"	"stato_doc_att"	
"mmmrsv"	"stato_doc_var"	

"mmidpassweb"	"id_sito_passweb"	
"mmest"	"id_valuta"	
"mmeve"	"cambio_vlt_euro"	
"mmdie"	"tp_cambio_vlt"	
"mmevc"	"cambio_euro_vlt"	
"mmdic"	"tp_cambio_euro"	
"mmydbt"	"gest_mydb_testa"	
"mmdatcre"	"data_creazione"	
"mmoracre"	"ora_creazione"	
"mmutecre"	"utente_crea"	
"mmmencre"	"menu_creazione"	
"mmdatvar"	"dt_var_sistema"	
"mmoravar"	"ora_var_sistema"	
"mmutemod"	"utente_modifica"	
"mmmenmod"	"menu_modifica"	
"mmosi"	"sigla_ordine"	
"mmose"	"serie_ordine"	
"mmonu"	"numero_ordine"	
"mmoda"	"data_ordine"	
"mmmsi"	"sigla_doc_orig"	
"mmmse"	"serie_doc_orig"	
"mmmnu"	"numero_doc_orig"	
"mmmda"	"data_doc_orig"	
"mmsigre"	"rif_sgl_attuale"	
"mmnumre"	"rif_nr_attuale"	
"mmdatre"	"rif_dt_attuale"	
"mmnrcroriga"	"id_riga"	
"mmtpr"	"tp_riga"	
"mmnte"	"id_rif_testata"	
"mmart"	"codice_articolo"	
"mmtsd"	"tp_descrizione"	
"mmdee"	"descr_riga"	
"mmdes"	"descr_articolo"	
"mmtip"	"tipo_articolo"	
"mmcol"	"nr_colli"	
"mmqta"	"quantita"	
"mmkta"	"coeff_conv"	
"mmtag"	"qta_taglia"	"mmtag" + "mmtagxl"
"mmord"	"tipo_stato_riga"	
"mmprz"	"prezzo"	
"mmsco"	"sconto"	
"mmsca"	"dt_sca_riga"	
"mmpro"	"perc_provv"	
"mmali"	"cod_iva"	
"mmtagtab"	"cod_taglia"	
"mmnrecroriga"	"id_riga_dba"	
"mmydbr"	"gest_mydb_riga"	
"mmspe"	"id_ind_sped"	
"mmanspe"	"cod_anag_sped"	
"mmtsr"	"tp_trasporto"	
"mmvet"	"cod_vettore"	
"mmpor"	"tp_porto"	
"mmdsp"	"descr_porto"	
"mmtps"	"tp_spese_sped"	
"mmvas"	"val_spese_sped"	
"mmaco"	"gest_colli_sped"	
"mmtco"	"nr_colli_sped"	
"mmape"	"gest_peso_sped"	
"mmtpe"	"peso_spedizione"	
"mmdtintr"	"dt_spedizione"	
"mmorintr"	"ora_spedizione"	
"mmpag"	"id_pagamento"	
"mmcbzpag"	"cod_banca"	
"mmacc"	"acconto_pagato"	
"mmspbanca"	"spese_banca"	

"mmspbolli"	"spese_bolli"	
"utenteum"	"utente_ult_mod"	
"dataum"	"data_ult_mod"	

## Prezzi Clienti / Articoli

"parttip"	"tipo_part"
"parttpc"	"tp_dato_conto"
"partcli"	"codice_conto"
"partcac"	"id_cat_conto"
"partzona"	"id_zona_conto"
"partstatcli"	"id_catsta_conto"
"parttpa"	"tp_dato_art"
"partart"	"cod_articolo"
"partcaa"	"id_cat_art"
"partstartcod"	"cod_catsta_art"
"partstartnum"	"nr_catsta_art"
"partnatura"	"cod_natura"
"partgrpmer"	"cod_grp_merc"
"partddt"	"data_inizio"
"partadt"	"data_fine"
"partiva"	"cod_iva"
"partval"	"id_valuta"
"parttpf"	"tp_fino"
"partcod"	"cod_articolo"
"partpromo"	"id_promozione"
"partappsu"	"tp_applica"
"partcondiz"	"tp_condizione"
"partappli"	"tp_applicazione"
"partfin"	"scaglione1"
"partprzn"	"part_1"
"parttp"	"tipo_pr_1"
"partagen"	"cod_agente_1"
"partcond"	"cod_cond_1"
"part_fin"	"scaglione_2"
"part_przn"	"part_2"
"part_tp"	"tipo_pr_2"
"part_agen"	"cod_agente_2"
"part_cond"	"cod_cond_2"
"utenteum"	"utente_ult_mod"
"dataum"	"data_ult_mod"

## Indirizzi di Spedizione

"dacin"	"id"
"dacto"	"cod_conto"
"dades"	"descrizione"
"daind"	"indirizzo"
"dacap"	"cap"
"daloc"	"localita"
"dapro"	"provincia"
"danaz"	"tp_nazionalita"
"dapae"	"cod_paese"
"datel"	"telefono"
"dafax"	"fax"
"damai"	"email"
"daint"	"url"
"dachi"	"chiusura_sett"
"dacon"	"riferimento"
"dacef"	"cell_rif"
"danot"	"nota"
"dadva"	"data_validita"

"dadcr"	"data_creazione"
"dadag"	"data_modifica"
"dacts"	"cod_conto_sost"
"dacis"	"id_ind_sost"
"daann"	"annullato"
"darag"	"gest_raggruppa"
"dacig"	"codice_cig"
"dacup"	"codice_cup"
"datpi"	"tp_indirizzo"
"utenteum"	"utente_ult_mod"
"dataum"	"data_ult_mod"

## Bolle Lavorazione

"obnum"	"codice"	
"obnso"	"cod_sottobolla"	
"obnsp"	"cod_sbl_padre"	
"obdat"	"dt_documento"	
"obmag"	"cod_mag_pf"	
"obmat"	"cod_mag_mp"	
"obsca"	"dt_scadenza"	
"obdst"	"nota"	
"obpia"	"gest_piano"	
"obcto"	"codice_conto"	
"obtip"	"tp_modu_stampa"	
"obsta"	"gest_stampa"	
"obval"	"tot_scarichi"	
"obinf"	"tp_info_pf"	
"obprz"	"gest_prezzi"	
"obass"	"fase_assegnata"	
"obdbp"	"cod_art_dbp"	
"obsml"	"cat_sml_bl"	
"obnbl"	"nr_righe_bl"	
"oltp"	"tp_riga"	
"ollav"	"codice_sbl_lav"	
"olart"	"codice_pf"	
"olpfp"	"codice_pf_padre"	
"olrif"	"nr_rif_oc"	
"olrii"	"nr_rif_impegni"	
"olcst"	"tot_corrente"	
"olcli"	"cod_cliente"	
"olspe"	"id_ind_sped"	
"oldbp"	"cod_dbp"	
"olsdo"	"serie_oc"	
"olord"	"codice_oc"	
"oldat"	"data_oc"	
"olsco"	"dt_scadenza_oc"	
"olsca"	"dt_scad_riga"	
"olntg"	"tab_taglie"	
"oldec"	"nr_decimali_qta"	
"olcol"	"nr_colli"	
"olqtf"	"quantita"	
"oltag"	"quantita_taglia"	"oltag" + "oltagxl"
"olcoe"	"coeff_conv"	
"olfsc"	"ult_fs_scarico"	
"olfsi"	"ult_fs_impegno"	
"olfsd"	"ult_fs_deposito"	
"olfst"	"ult_fs_stampata"	
"olcbl"	"cod_sbl_lavora"	
"olcfs"	"prec_fs_s_sbll"	
"olmag"	"cod_mag_carico"	
"olnrx"	"nr_record"	
"olval"	"val_corrente_s"	
"oluni"	"nr_univoco_riga"	

"olunp"	"nr_rif_pf_padre"	
"ollot"	"id_lotto"	
"olare"	"codice_area"	
"olcom"	"codice_commissa"	
"oltdo"	"cod_sigla_doc"	
"olstr"	"stato_riga"	
"olnot"	"nota_riga"	
"olsmplf"	"categ_sml_pf"	
"obcrs"	"gest_no_sml_sps"	
"obdtc"	"dt_creazione"	
"olric"	"nr_rif_crea"	
"utenteum"	"utente_ult_mod"	
"dataum"	"data_ult_mod"	

## Distinta Base

"tpcod"	"codice"	
"tpsvi"	"cod_art_costo"	
"tpctr"	"gest_ctrl_disp"	
"tpfpf"	"codice_pf_padre"	
"tpagg"	"cod_tab_abbina"	
"tpsmi"	"categoria_sml"	
"tpusa"	"gest_dbvc_dbvo"	
"tpcol"	"gest_colleg_sml"	
"tpump"	"tp_um_usata"	
"tpmmp"	"cod_mag_mp"	
"tpdec"	"nr_decimali_qta"	
"tpcli"	"cod_cliente"	
"tpoap"	"nr_ore_appronta"	
"tpore"	"nr_ore_lotto"	
"tpnr"	"qta_lotto_prod"	
"tpkoe"	"coeff_conv"	
"tpkoc"	"consumo_taglia"	"tpkoc" + "tpkocxl"
"tpdes"	"descr_fase"	
"tpfsc"	"tp_cond_fase"	
"tpult"	"descr_agg_fase"	
"tpnqt"	"qta_rif_fase"	
"tpdoc"	"doc_stampa_fase"	
"tpnot"	"note_fase"	
"tplag"	"wip_fase"	
"tptyp"	"lag_fase"	
"tpfas"	"nr_fase"	
"tpnfasi"	"nr_tot_fasi"	
"tpctl"	"gest_clav_fase"	
"tpncomp"	"nr_cmp_fase"	
"tpind"	"indice_cmp"	
"tpart"	"codice_mp"	
"tpqta"	"qta_utilizzo"	
"tpcoe"	"coeff_conv_mp"	
"tpevi"	"tp_evidenza"	
"tpnec"	"nega_cond_cli"	
"tpned"	"nega_cond_data"	
"tpnet"	"nega_cond_tg"	
"tpsss"	"cod_art_sost"	
"tpmag"	"cod_mag_impegni"	
"tpneg"	"nega_cond_art"	
"tpcon"	"tp_cond_art"	
"tpcar"	"cod_art_cond"	
"tpunm"	"nr_unita_misura"	
"tpccl"	"cod_cli_cond"	
"tpcdd"	"dt_inizio_cond"	
"tpcda"	"dt_fine_cond"	
"tpcno"	"nota_riga_01"	
"tpcno1"	"nota_riga_02"	

"tpcno2"	"nota_riga_03"	
"tpcno3"	"nota_riga_04"	
"tpcno4"	"nota_riga_05"	
"tpcno5"	"nota_riga_06"	
"tpcno6"	"nota_riga_07"	
"tpcno7"	"nota_riga_08"	
"tpcno8"	"nota_riga_09"	
"tpcno9"	"nota_riga_10"	
	"nota_riga_11"	
	"nota_riga_12"	
	"nota_riga_13"	
	"nota_riga_14"	
	"nota_riga_15"	
	"nota_riga_16"	
	"nota_riga_17"	
	"nota_riga_18"	
	"nota_riga_19"	
	"nota_riga_20"	
	"nota_riga_21"	
	"nota_riga_22"	
	"nota_riga_23"	
	"nota_riga_24"	
	"nota_riga_25"	
"tptag"	"condizione_tg"	
"tpcts"	"tp_sost_cmp"	
"tpdir"	"tp_dir_ricerca"	
"tplot"	"gest_spezza_qta"	
"tpede"	"gest_escludi_dl"	
"tpesc"	"gest_escludi_sl"	
"tpest"	"gest_esterna"	
"tpcst"	"tp_costo_mp"	
"tpcos"	"cos_lavorazione"	
"tpefo"	"cod_fornitore"	

## Impegni

"ilndo"	"codice"	
"ilnsd"	"cod_sottobolla"	
"ilnrb"	"nr_rif_pf"	
"ilfas"	"nr_fase"	
"ilmag"	"cod_magazzino"	
"ilamp"	"cod_articolo"	
"ilidubi"	"id_ubicazione"	
"ildad"	"dt_sca_riga_bl"	
"iltqt"	"tab_taglie"	
"ilcol"	"nr_colli"	
"ilqta"	"quantita"	
"ilqtg"	"quantita_taglia"	"ilqtg" + "ilqtgxl"
"ilprz"	"prezzo"	
"ilevi"	"tp_evidenza"	
"ilind"	"indice"	
"ildep"	"gest_depositato"	Vale S/N invece di 1/0
"iledl"	"gest_escludi_dl"	Vale S/N invece di 1/0
"ilesl"	"gest_escludi_sl"	Vale S/N invece di 1/0
"ilgcst"	"gest_costo_dbp"	Vale S/N invece di 1/0
"il_mxqtlot"	"nr_righe_lotto"	
"ill_idlot"	"id_lotto"	
"ill_ndqta"	"nr_decimali_qta"	
"ill_lqta"*	"qta_lotto"	
"ill_lcolli"	"nr_colli_lotto"	
"ill_lqtatgl"	"qta_tg_lotto"	
"utenteum"	"utente_ult_mod"	
"dataum"	"data_ult_mod"	

